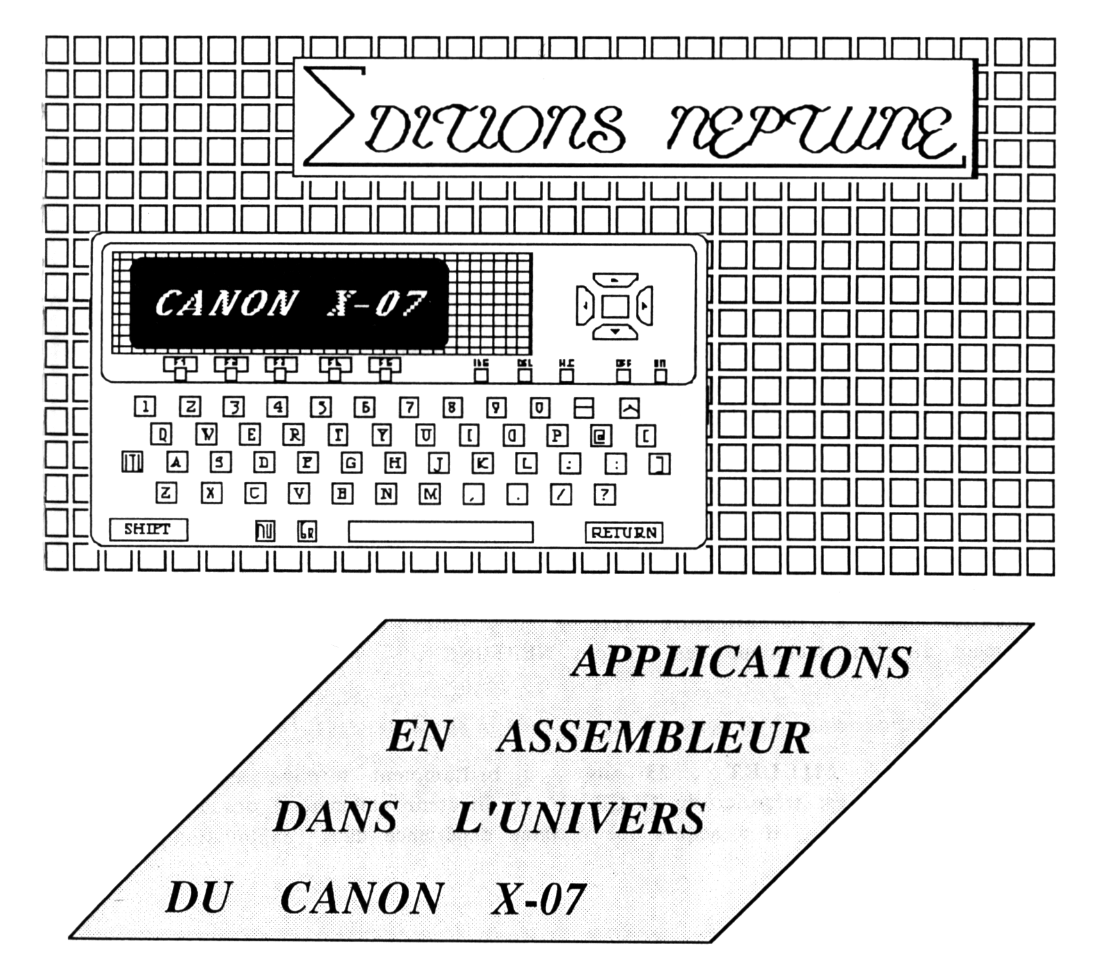
André TONIC
Edward AREVIAN
Philippe MILLET
Jean-Jacques ROUSSEAU

APPLICATIONS EN ASSEMBLEUR DANS L'UNIVERS DU CANON X -07





APPLICATIONS EN ASSEMBLEUR DANS L'UNIVERS DU CANON X-07

par

André TONIC

Edward AREVIAN

Philippe MILLET

Jean-Jacques ROUSSEAU

18 bis RUE VIOLET , 75015 PARIS

Tél : 64.56.81.56

LES AUTEURS

André TONIC , 19 ans , étudiant en Sciences Economiques , participe activement depuis plusieurs années à l'élaboration de nombreux logiciels commerciaux sur micro-ordinateurs aussi bien portables qu'individuels . En outre , il est co-auteur de plusieurs recueils de programmation . Il est également président du CLUB C7 et directeur de la publication des Editions NEPTUNE .

Edward AREVIAN , 23 ans , étudiant en électronique et en informatique, est conseiller technique au sein du CLUB C7 . Il s'intéresse tout particulièrement à l'architecture interne des micro-ordinateurs . Il est également directeur technique des Editions NEPTUNE .

Philippe MILLET , 23 ans , a brillamment terminé ses études commerciales : il est trésorier du CLUB C7 et directeur commercial des Editions NEPTUNE . De plus , il a acquis une certaine expérience dans l'exploitation des langages évolués .

Jean-Jacques ROUSSEAU , 43 ans , est maitre de conférences à l'université du Maine . Il y enseigne la cristallographie et l'electronique . Passionné d'informatique , il est l'auteur de très nombreux logiciels écrits aussi bien en BASIC qu'en ASSEMBLEUR . De plus , il a acquis une très grande connaissance des parties internes du X-07 .

REMERCIEMENTS

Nous désirons adresser **nos plus vifs remerciements** à :

_ **Mrs De la Rüe du Can et De Carville** , responsables MARKETING de la société CANON FRANCE , pour leurs informations relatives au X-07 .

_ **Melle HELAS** , étudiante , pour sa participation à la composition de cet ouvrage .

Les AUTEURS

AVANT - PROPOS

Composé dans la continuité des "Mystères du X-07" , ce deuxième ouvrage complète les connaissances acquises lors de la lecture de ce dernier . Bâti autour de trois grandes parties (Soft , Technique , Applications) , ce livre constitue la première référence en matière d'applications pures dans l'univers du X-07 .

Tout en reprenant les recettes qui ont fait le succès des "Mystères" (Taille, clarté , exemples , schémas ...) , "Applications en ASSEMBLEUR dans l'univers du CANON X-07" apporte son lot de nouveautés avec une présentation originale, un style différent et surtout une pléiade d'applications qui satisferont le plus exigeant des canonistes .

PREFACE

AVRIL 1989 , KOUROU , GUYANE FRANCAISE ...

Il faisait très chaud au centre d'essais depuis quelques heures .. Pourtant ce n'était pas dû aux rayons du soleil , cachés depuis quelques temps derrière d'épais nuages laiteux . Non ... L'explication était toute autre et seuls les initiés pouvaient comprendre l'énorme enjeu qui se jouait devant eux .

Tout à coup , une énorme flamme bleue se profila sur les écrans de contrôle , suivie d'une fine volute blanche annonciatrice de réjouissances ... **Les moteurs d'HERMES , première navette européenne , venaient de s'allumer !!** Un sans-faute pour toute l'équipe de ce projet , qui travaillait d'arrache-pied depuis dix ans .
La spectaculaire destruction , en Janvier 1986 , de la navette CHALLENGER avait sensiblement refroidi les ardeurs américaines . Les yeux du monde entier s'étaient alors tourné vers l'Europe ... vers HERMES !

En ce 24 avril de l'année 1989 , une page de l'histoire technologique du vieux continent était tournée ... HERMES était fin prête !

JUIN 1989 , PARIS ...

Tout en marchant le long des couloirs de ROISSY , j'ai l'intuition que mon séjour à Kourou ne va pas être une partie de plaisir ... Vérifier tous les systèmes extérieurs d'HERMES ne va pas être de tout repos !

Pourtant , à 25 ans , j'ai réussi à me hisser au poste d'ingénieur informaticienne alors qu'au départ , certains me destinaient plus volontiers à une carrière de mannequin ...

Pourquoi ai-je été choisie alors que d'autres apparaissaient plus compétents ? "Tout simplement" à cause de **mon X-07** ! Que vient faire cette machine dans cette histoire ? ...

Il y a un peu moins d'un an , j'ai réalisé **un système de détection de pannes "dernier cri"** : détecteur ultra-perfectionné en relation infrarouge constante avec les ordinateurs centraux , rapidité de travail phénoménale , X-07 comme "cerveau principal" dopé avec deux Méga-octets de mémoire et quelques processeurs ...

Ce genre de détecteur "tout terrain" commençait à devenir très apprécié dans les technologies de pointe ... Il suffisait de faire ingurgiter au X-07 les caractéristiques d'une panne et le détecteur se chargeait du reste . Dès qu'une panne était constatée , l'ordinateur central prenait le relais en affinant les résultats .

Ce projet avait été ma carte de passage ... Espérons qu'il ne me fera pas faux bond en Guyane !

"Poste 3 ... Boeing 747 à destination de Kourou , Guyane ... Embarquement immédiat ..." . C'est pour moi ! Un dernier coup d'oeil à Paris et me voilà dans les nuages ... Tout en m'enfonçant confortablement dans mon siège , je commence à vérifier fébrilement tout mon matériel . Je prends **mes deux "bibles X-07" ("Les mystères du X-07" et "Applications en ASSEMBLEUR dans l'univers du CANON X-07")** et me voilà parti pour quatre heures de minutieuses mises au point ...

20 AOUT 1989 , KOUROU , GUYANE ...

Depuis deux mois , je n'arrête pas ! Je dors à peine quatre heures par nuit mais je ne suis pas fatiguée . L'excitation du départ prochain d'HERMES me donne des ailes ...

Mes vérifications techniques se sont bien passées . Le X-07 a merveilleusement bien fonctionné et une panne de compresseur a été détectée, il y a deux jours . La réparation est terminée depuis trois heures et je m'escrime à y déceler une éventuelle défaillance ... Mais apparemment , rien d'anormal !

Le "jour J" approche ... Dans cinq jours , nous serons fixés définitivement sur l'avenir de l'Europe spatiale .

25 AOUT 1989 . KOUROU . PAS DE TIR ...

C'est l'effervescence au centre de contrôle ! Tout le gouvernement français est là ... Le président a l'air particulièrement fébrile .

Le compte à rebours est commencé depuis pas mal de temps déjà ... Il devrait rester environ trois minutes .

L'excitation monte dans la salle et les yeux d'une centaine de techniciens sont braqués sur les indicateurs de panne . Aucun d'eux ne semble vouloir s'allumer . Le CANON aurait-il détecté toutes les pannes ? ...

Ca y est ! Il reste cinq secondes .... quatre .... trois .... deux .... une .... zéro ... **Mise à feu des moteurs !!**

HERMES s'élève majestueusement du sol , déroulant une magnifique trainée blanche ... Elle poursuit sa route comme prévu et **se place en orbite** quelques instants plus tard !

C'est le délire à Kourou !! Chacun a droit à une part de cet énorme succès . Toutes les personnes présentes sont félicitées personnellement par le président de la nation . En s'approchant de moi , il me souffle : " Comment avez-vous fait pour vérifier tous les systèmes en si peu de temps ? ..." . Je lui réponds malicieusement : "Vous savez , ce n'est pas compliqué .... Il suffit de deux bons livres de recettes , d'un X-07 et de quelques condiments . Vous y rajoutez un peu de jugeotte et le tour est joué !" .

Nous rions ensemble en regardant HERMES s'épanouir dans l'espace interplanétaire ... **Bravo l'Europe et merci CANON !**

A. TONIC

INTRODUCTION

Après le grand succès des **"Mystères du X-07"** , nous nous devions absolument d'écrire un second ouvrage , une suite ...

En effet , après avoir appris l'ASSEMBLEUR du X-07 , tous les utilisateurs de cette merveilleuse machine ont ardemment désiré aller plus loin , mieux connaître l'interface X-720 , la mémoire , les ports de sortie , l'utilisation de la zone système , que sais-je encore ? ...

Pour toutes ces personnes et pour tous ceux qui adorent utiliser des logiciels "clés en main" , nous avons repris notre plume en compagnie d'un nouveau technicien , **Jean-Jacques ROUSSEAU** , qui arbore un nom tout à fait prédestiné !!

Avant tout chose , je souhaiterais souligner **un point primordial** : cet ouvrage s'appuie en partie sur les connaissances que vous devez avoir acquises à la lecture des "Mystères du X-07" . Par conséquent , si vous ne l'avez pas encore lu , mettez au chaud le livre que vous tenez entre les mains et compulsez attentivement le premier ouvrage . Cela vous sera très utile et vous aidera à mieux assimiler les notions exposées ci-après .

"Applications en ASSEMBLEUR dans l'univers du CANON X-07" est constitué principalement d'applications , apparemment très hétérogènes , mais toutes reliées au moins par un point commun : elles sont toutes écrites en ASSEMBLEUR X-07 et exploitent à fond les possibilités du CANON .

Nous vous conseillons de commencer à lire cet écrit à 8 heures du matin et non à 8 heures du soir car sinon , votre nuit sera tellement mouvementée que vous maudirez le ciel de ne pas avoir acheté un matelas "Epéda-multispires" !!!

Après ces petites recommandations , entrons dans le vif du sujet ... Que contient ce palpitant assemblage de feuilles ?

Juste après le sommaire , nous allons vous emporter dans un tourbillon d'octets , d'adresses , **de langage machine** . En fait , vous apprendrez à manipuler , à contorsionner , à réduire , à fouiller la mémoire de votre X-07 comme vous ne l'avez encore jamais fait ! ... C'est fou ce que la petite mémoire du CANON peut contenir comme surprises ...

Ensuite , après avoir bu un petit soda et vous être reposé quelques instants , vous pourrez poursuivre votre quête ... Quelques chapitres **plus techniques** vous emporteront dans le monde obscur de la communication et des périphériques .

Enfin , après tous ces fascinants périples , vous désirerez probablement vous détendre . A cet effet , nous vous avons concocté **une dizaine de logiciels** tous plus intéressants les uns que les autres .

Le soleil se couchera sur l'horizon auréolé de brume quand vous terminerez la lecture **des annexes** ... Vous tomberez de sommeil en pensant : "Ah !! ... Si je n'avais pas connu les Editions NEPTUNE et le CLUB C7 , que serions-nous devenu , moi et mon X-07 ? ..." .

A. TONIC

SOMMAIRE

1ère PARTIE : Où l'on commence par du SOFT ...

2ème PARTIE : ... Où l'on continue par du HARD ...

3ème partie : ... Et où l'on termine "LOGICIEL" !

ANNEXES

1ère PARTIE :

Où l'on commence par du SOFT ...

UTILISATION de ROUTINES

Nous allons débuter cette première partie par l'un des problèmes les plus importants auquel se heurtent la plupart des programmeurs : **l'utilisation des sous-programmes en langage machine sous BASIC** .

Comme vous l'avez sans doute remarqué quand vous programmiez , l'un des défauts majeurs du BASIC est **sa grande lenteur d'exécution** . On peut souvent améliorer de façon importante la vélocité d'un logiciel ... Comment ? Tout simplement en écrivant en ASSEMBLEUR les parties critiques du programme !

Deux commandes du BASIC MICROSOFT du X-07 permettent la communication entre les deux langages (BASIC et L.M.) .

D'une part , l'instruction **"EXEC adresse"** permet de lancer une routine écrite en ASSEMBLEUR à partir du BASIC . Notons que cette commande possède une lacune majeure : elle ne permet pas le passage d'arguments .

D'autre part , la directive **"USR (adresse , argument)"** permet également de lancer un sous-programme écrit en langage machine . L'intérêt majeur de cette commande réside dans le fait qu'elle autorise le passage d'arguments .

Tout d'abord , examinons ensemble le problème de la mise en place de la routine ...

1.1 ADRESSE D' IMPLANTATION D' UNE ROUTINE L.M.

Après chaque arrêt et remise en route normale (sans passer par l'instruction SLEEP) du X-07 , les pointeurs de début et de fin de la zone BASIC sont réactualisés . Nous rappelons que le rôle de ces pointeurs et leur utilisation sont décrits dans "les MYSTERES du X-07" : nous ne reviendrons donc pas dessus .

Par conséquent , il est donc possible d'introduire , sans risques importants, **des routines écrites en ASSEMBLEUR après le logiciel BASIC** .

La méthode classique consiste à introduire les codes dans la zone réservée au moyen d'un ordre CLEAR (Exemple : CLEAR 50,4000 fixe la limite de la zone utilisateur à l'adresse 4000 . Donc , les octets situés après l'adresse 4000 peuvent être utilisés pour écrire des routines L.M.) .

La zone réservée est caractérisée par deux pointeurs . **MEMSIZ** (adresse &H01DF en hexadécimal) représente le pointeur de début (fin de la zone BASIC) et se trouve être initialisé par la commande CLEAR . Le pointeur de fin de cette zone se nomme **RAMSTRT** (adresse &H210) et fournit le début précis de la zone "fichiers RAM" . Ce pointeur est donc actualisé par l'instruction FSET.

Il existe une deuxième méthode pour implanter vos routines . **Il suffit simplement d'écrire votre sous-programme dans le fond de la zone des fichiers RAM** . Bien évidemment , il faut disposer d'une place suffisante dans cet espace . L'ordre **DIR** permet d'effectuer ce contrôle .

Si on utilise cette éventualité , il n'est plus nécessaire d'exécuter l'ordre CLEAR à chaque mise sous tension du CANON . Par contre , il est primordial de veiller à ne pas écraser la routine en écrivant dans la zone RAM . De la même façon , vous devez éviter d'implanter votre routine sur un logiciel stocké comme "fichier RAM" .

Ces deux méthodes imposent au programmeur de contrôler **la gestion de la mémoire** et les adresses d'implantation dépendant de la version utilisée (CANON 8 Ko , 12 Ko , 16 Ko , 20 Ko , 24 Ko ... ou plus maintenant grâce aux sociétés de service : pour plus de renseignement à ce sujet , vous pouvez consulter les gazettes du CLUB C7) .

Notons qu'il existe d'autres possibilités permettant de s'affranchir de ces difficultés ... Mais il est vrai que d'autres problèmes apparaissent ! Nous vous les présentons ci-dessous .

1.1.1 La méthode de l'instruction "REM" .

L'interpréteur ignore ce qui se trouve après une instruction "REM" . En effet , quand un logiciel est en cours d'exécution , dès que l'interpréteur découvre "REM" , il passe automatiquement à la ligne suivante . Il est donc tout à fait possible d'implanter une routine après REM .

L'utilisation de cette méthode est relativement aisée . Il faut tout d'abord créer une ligne contenant l'intruction REM , suivie d'un nombre de caractères égal à la longueur de la routine . Ensuite , vous pouvez implanter entièrement votre routine par la commande **"POKE adresse , code"** .

Il s'avère très facile de déterminer l'adresse du début d'implantation grâce à l'instruction **"RESTORE n° de ligne"** . En effet , cette directive charge dans les **octets &H328 et &H329** l'adresse absolue du début de la ligne en question . Cette adresse est donc égale à :

AD = PEEK (&H329) * 256 + PEEK (&H328) + 6

Il faut impérativement rajouter le chiffre 6 afin de tenir compte des octets de chaînage , du numéro de ligne et du code de l'instruction REM (Voir les "MYSTERES du X-07" , pages 87 à 90) .

Cette possibilité possède plusieurs avantages très intéressants : tous les codes sont admissibles , l'implantation est absolue et invariante , le code est résident . De ce fait , chaque logiciel BASIC apporte ses propres routines L.M. et l'utilisateur n'a pas à s'occuper du chargement de codes supplémentaires .

Par contre , la longueur d'une routine est limitée à **80 codes sur l'écran LCD et à 255 sur l'écran vidéo** . De plus , certains codes peuvent "troubler" l'interpréteur : il est donc vivement conseillé de sauter la ligne "truffée" d'une routine au moyen de l'instruction "GOTO" . Enfin , l'édition de la ligne , après l'implantation , est impossible et les résultats du listage sont imprévisibles ... (Voir l'exemple 1) .

1.1.2 La méthode de la chaîne de caractères .

Dans le cadre du BASIC MICROSOFT , il existe deux possibilités d'implantation d'une chaîne de caractères ...

1.1.2.1 Dans le corps du programme .

Par exemple , la chaîne contenue dans la ligne 10 A$="TURLUTUTU" possède une adresse fixe étant donné qu'elle réside dans le corps du programme . Cette adresse peut être obtenue par l'intermédiaire de la fonction VARPTR :

AD = VARPTR (A$) : AD = PEEK (AD + 2) * 256 + PEEK (AD + 1)

La routine peut alors être implantée grâce à l'instruction "POKE" dans la chaîne de caractères .

Les avantages sont constitués par l'adresse d'implantation absolue et le code résident .

Par contre , les résultats de listage sont totalement imprévisibles . D'autre part , **le code 0** est interdit car il indique une fin de ligne . De même , **le code &H22** est à bannir : il représente le guillemet , donc une fin de chaîne ... Nous vous laissons imaginer les dégats prévisibles !!

1.1.2.2 Dans la zone des chaînes .

Par exemple , la chaîne résidant dans la ligne 10 A$ = STRING$ (100 , "*") est contenue dans la zone des chaînes de caractères . Son adresse n'est donc pas absolue et peut évoluer tout au long de l'exécution du programme .

Notons que tous les codes peuvent être utilisés et que les problèmes de listage disparaissent .

Le défaut majeur de cette méthode est constitué par le fait qu'elle n'est utilisable qu'avec **les programmes relogeables** . De plus , il faut impérativement recalculer l'adresse d'implantation avant chaque appel à la routine . D'autre part , il est obligatoire de réimplanter la routine à chaque utilisation (Voir exemple 2) .

1.1.3 La méthode du tableau d'entiers .

Avec cette possibilité , le code machine est introduit dans un tableau d'entiers (Deux codes pour chaque valeur d'indice) dont on récupère l'adresse en utilisant **la fonction VARPTR (T(0))** .

Cette méthode est délicate à mettre en oeuvre si la longueur de la routine est importante mais elle offre l'avantage de permettre le passage aisé d'arguments multiples . Quelques précautions sont bien évidemment à prendre lors du codage de la routine . En effet , les codes sont introduits deux par deux et sont croisés . Voici un petit exemple :

"Source" normal	"Source" modifié	Code résultant
	NOP	00
LD HL , source	LD HL , source	21 xx xx
	NOP	00
LD DE , destination	LD DE , destination	11 yy yy
	NOP	00
LD BC , longueur	LD BC , longueur	01 zz zz
LDIR	LDIR	ED B0
RET	RET	C9
	NOP	00

Les trois premiers "NOP" servent au cadrage . Le dernier permet d'avoir un nombre pair d'octets .

Voici l'implantation résultante :

	Codes normaux	Codes croisés	Valeur décimale
EX (0)	00 21	21 00	8448
EX (1)	(source)	codage effectué par l'interpréteur du X-07	
EX (2)	00 11	11 00	4352
EX (3)	(destination)	codage effectué par l'interpréteur du X-07	
EX (4)	00 01	01 00	256
EX (5)	(longueur)	codage effectué par l'interpréteur du X-07	
EX (6)	ED B0	B0 ED	- 20234
EX (7)	C9 00	00 C9	201

Cette méthode a effectivement quelques petits inconvénients : difficulté de mise en oeuvre , utilisation uniquement avec des codes relogeables , évolution de l'adresse d'implantation .

En conclusion , nous pouvons remarquer **la relative multiplicité des méthodes d'implantation de routines L.M** . Notons qu'après la mise au point du programme , il est possible de supprimer les parties du logiciel ayant servi à l'implantation de la routine (uniquement pour les méthodes 1.1.1 et 1.1.2.1) .

1.2 APPEL DE LA ROUTINE .

Nous abordons un secteur très important de la programmation en ASSEMBLEUR : **l'appel d'une routine à partir du BASIC** . Voici , ci-dessous, les différentes méthodes ...

1.2.1 Sans passage d'arguments .

L'appel d'une routine sans passage d'arguments constitue le cas le plus simple ... Il suffit d'utiliser la fonction **"EXEC adresse"** . Après exécution de la routine L.M. , la commande EXEC fait pointer l'interpréteur sur l'instruction suivante . Si la routine ne se termine pas par **l'instruction "RET"** de l'ASSEMBLEUR Z-80 , le CANON se bloque et le "RESET" est nécessaire ...

1.2.2 Avec passage d'arguments .

1.2.2.1 Argument unique .

Si l'argument est unique , on doit utiliser **Z = USR (adresse , argument)** . Deux cas sont alors possibles ...

Si l'argument est **une chaîne de caractères** , le registre **DE** contient l'adresse (ADR) dès l'entrée dans la routine . De plus , l'octet pointé par ADR contient la longueur de la chaîne .

L'adresse physique de la chaîne est contenue dans les deux octets pointés par les adresses ADR + 1 et ADR + 2 .

Si l'argument est **un nombre** , la routine USR transfère la valeur de l'argument dans **l'accumulateur (&H44E à &H455)** avant d'entrer dans le sous-programme . De plus le registre A contient le type de donnée (2=entier , 4=simple précision , 8=double précision) et le registre HL pointe sur le début de l'accumulateur (&H44E) .

Pour récupérer l'argument , on peut utiliser les routines de transfert de données (Voir les "MYSTERES ..." , pages 105-106) . Par exemple , **la routine CA26** tranfère le contenu de l'accumulateur dans la paire de registres BC/DE (cas des nombres en simple précision) .

Pour le transfert avec des entiers , vous pouvez utiliser la routine suivante :

```
INC HL
INC HL
PUSH HL        ; sauvegarde de HL pour le retour de la routine
                 (inutile s'il n'y a pas d'argument à retourner au BASIC)
LD E,(HL)
INC HL
LD D,(HL)
EX DE,HL       ; HL contient l'argument entier passé par la fonction USR
```

Pour le retour , la routine USR effectue l'opération inverse : le contenu de l'accumulateur est transféré dans la variable d'appel de USR .

Dans le cas des entiers , vous pouvez passer par l'intermédiaire de la méthode décrite à la page suivante ... On suppose que l'entier à retourner est contenu dans le registre BC :

```
POP HL          ; récupère l'adresse des entiers dans l'accumulateur
LD (HL),C
INC HL
LD (HL),B
RET             ; retour au BASIC
```

1.1.2.2 Arguments multiples .

Si vous avez plusieurs arguments à faire passer , **la méthode des tableaux est très puissante** . Si elle n'est pas utilisable , il faut passer les arguments au moyen d'un tableau dont l'adresse sera l'argument de la routine USR .

Voici un exemple précis ... On désire passer les arguments suivants contenus dans le tableau d'entiers :

ARG (0) = aa00
ARG (1) = xxyy
ARG (2) = dd00
ARG (3) = zzuu

L'appel de la routine se fait par **T = USR (adresse,VARPTR (ARG(0))).**

Début de la routine :

```
INC HL
INC HL
PUSH HL     ; HL pointe l'adresse de ARG (0)
LD E,(HL)
INC HL
LD D,(HL)
PUSH DE     ; DE = adresse absolue de ARG (0)
POP IX
```

On obtient alors (IX+0) = aa (IX+1) = 00 (IX+2) = xx (IX+3) = yy

Par conséquent , l'adressage indexé est utilisé afin de récupérer les valeurs des arguments .

Pour le retour , on peut utiliser la même technique .

Remarquons que dans le cas d'une implantation en adresse absolue de la routine , les arguments seront envoyés par l'instruction POKE avant l'appel de la routine .

Après ces explications théoriques , nous vous avons concocté **quelques exemples supplémentaires** qui vont vous aider à assimiler les importantes notions que nous venons d'aborder ...

1.3 EXEMPLES D' UTILISATION .

1.3.1 Exemple N°1 .

Ce premier exemple va permettre de concrétiser **la méthode de l'instruction REM** que nous venons de voir (paragraphe 1.1.1) .

Remarquons une astuce très intéressante : toute erreur survenant après la ligne 50 provoque **le passage automatique dans le mode d'édition LIST @** .

Voici le listing du programme :

```
10 GOTO 20
15   REM01234567890123456789
20 RESTORE 15 : AD% = PEEK (&H329) * 256 + PEEK (&H328) + 6
30 FOR I% = 0 TO 19 : READ B$ : POKE AD% + I% , VAL ("&H" + B$): NEXT I%
40 DATA 06 , 08 , E1 , 10 , FD , 2A , 15 ,03 , E5 , D1 , CD , 0B , F3 , FD , 21 ,
    00 , 40 , C3 , 8A , FE
50 ON ERROR GOTO 65000
60 ' DEBUT DU PROGRAMME
100 A = 5 : B = 0 : C = A/B
64999 END
65000 PRINT "Erreur"
65010 EXEC AD%
```

1.3.2 Exemple N°2 .

Ce second exemple va démontrer l'efficacité de **la méthode de la chaîne de caractères** (paragraphe 1.1.2.2) .

Voici son listing :

```
10 A$ = STRING$ (20,"#")
20 AD% = VARPTR (A$) : AD% = PEEK (AD% + 1) + 256 * PEEK (AD% + 2)
30 FOR I% = 0 TO 19 : READ B$ : POKE AD% + I% , VAL ("&H" + B$): NEXT I%
40 DATA 06 , 08 , E1 , 10 , FD , 2A , 15 , 03 , E5 , D1 , CD , 0B , F3 , FD , 21 ,
    00 , 40 , C3 , 8A , FE
50 ON ERROR GOTO 65000
60 ' DEBUT DU PROGRAMME
100 A = 5 : B = 0 : C = A/B
64999 END
65000 PRINT "Erreur"
65005 AD% = VARPTR (A$) : AD% = PEEK (AD% + 1) + 256 * PEEK (AD% + 2)
65010 EXEC AD%
```

1.3.3 Exemple N°3 .

La méthode des tableaux d'entiers va être explicitée ci-dessous . Etudiez-la consciencieusement ... Bien qu'un peu compliquée , elle pourra vous rendre de nombreux services !!

Voici son listing :

```
10 DIM U%(9)
20 U%(0) = &H0806 : U%(1) = &H10E1 : U%(2) = &H2AFD : U%(3) =
     &H0315 : U%(4) = &HD1E5
30 U%(5) = &H0BCD : U%(6) = &HFDF3 : U%(7) = &H0021 : U%(8) =
     &HC340 : U%(9) = &HFE8A
50 ON ERROR GOTO 65000
100 D$ = 5
64999 END
65000 PRINT "Erreur"
65010 EXEC VARPTR (U%(0))
```

1.3.4 Exemple N°4 .

Ce quatrième exemple expose **la manière d'écrire une même valeur dans toutes les cases d'un tableau à une dimension** .

Si ce tableau est composé d'entiers , on aura U(5) = N*2 . Par contre , si le tableau est composé de réels en double précision , on aura U(5) = N*8 .

Vous pouvez comparer la vitesse d'exécution de cette petite routine avec celle d'une boucle écrite en BASIC ... Une légère différence transparaît !!

Voici le listing :

```
10 DEFINT A-Z : N = 50 : DIM U(7) , A!(N)
15 AD = 0
20 U(0) = 8448 : U(2) = 4352 : U(4) = 256 : U(6) = -20243 : U(7) = 201
30 A!(0) = 100 : GOSUB 100
40 PRINT A!(0) ; A!(1) ; A!(20)
50 A!(0) = 0 : GOSUB 100
60 PRINT A!(0) ; A!(1) ; A!(20)
70 END
100 U(1) = VARPTR (A!(0)) : U(3) = VARPTR (A!(1)) : U(5) = N*4
110 EXEC VARPTR (U(0)) : RETURN
```

1.3.5 Exemple N°5 .

Cet avant-dernier exemple vous expose **la façon de remplir un écran vidéo très rapidement** .

Vous devez ajuster la valeur de la variable US(5) selon le mode écran . Notons que le code à reproduire se trouve en ligne 40 .

Voici le listing de cette routine :

```
5 SCREEN 2 : CLS
6 LINE (0,0) - (200,180)
10 DEFINT A-Z : DIM US(7)
20 US(0) = 8448 : US(2) = 4352 : US(4) = 256 : US(6) = -20243 : US(7) =
   201
30 US(1) = &H8000 : US(3) = &H8001 : US(5) = 511
40 POKE &H8000 , 64
50 EXEC VARPTR (US(0))
70 END
```

1.3.6 Exemple N°6 .

Si la fonction **"RESTORE n° de ligne"** est présente dans le BASIC du CANON, la fonction **"RESTORE variable"** provoque , par contre , un singulier "UL Error" .

Cette fonction peut être très facilement implémentée au moyen de la petite routine relogeable incluse dans le listing ci-joint .

On commence par charger **le registre DE** avec la valeur de la variable . Ensuite , **la routine F30D** (Voir les "MYSTERES" , page 109) recherche le numéro de ligne BASIC et en retour , la valeur de l'adresse de la ligne se trouve dans **le registre BC** . Notons que si la ligne n'est pas trouvée , le message "UL Error" apparaît tout à fait normalement . Finalement , le pointeur des DATA est actualisé .

Dans le listing ci-dessous , la partie supérieure du logiciel "LOGOGENESE" présent dans le chapitre des applications (en fin d'ouvrage) est reprise . En effet , ce logiciel utilise l**a technique du RESTORE calculé** . Normalement , 33 lignes de DATA sont présentes des lignes 100 à 132 ainsi que 14 autres des lignes 200 à 213 .

Notons que la routine Z = USR (AD,J) est équivalente à RESTORE J . De plus, la ligne 90 calcule l'adresse d'implantation de la chaîne au moment de l'appel de la routine .

Voici le listing :

```
1 ' ATTENTION : NE PAS RENUMEROTER !!
10 DEFINT A-Z : CLS : PRINT "Logogénèse"
20 RM$ = STRING$ (18,0) : GOSUB 90
30 FOR I = 0 TO 17 : READ B$ : POKE AD + I , VAL ("&H" + B$) : NEXT :
   Z = RND (0)
40 DATA 23 , 23 , 5E , 23 , 56 , CD , 0D , F3 , 60 , 69 , D2 , 38 , F6 , 2B , 22 ,
    28 , 03 , C9
45 J = INT (RND (1) * 33) + 100 : GOSUB 90
50 Z = USR (AD,J) : N = INT (RND (1) * 4 + 1)
55 FOR I = 1 TO N : READ D$ : NEXT
60 K = INT (RND (1) * 14) + 200 : GOSUB 90
65 Z = USR (AD,K) : N = INT (RND (1) * 4 + 1)
70 FOR I = 1 TO N : READ F$ : NEXT
75 PRINT D$ ; F$ : IF INKEY$ = "" THEN 45
80 IF INKEY$ = "" THEN 80 ELSE 45
90 AD = VARPTR (RM$) : AD = PEEK (AD + 1) + 256 * PEEK (AD + 2) :
   RETURN
95 END
```

Si vous désirez plus de compléments sur cette technique de "RESTORE calculé" , vous pouvez utilement consulter la gazette N°5 du CLUB C7 ...

Après ces quelques exemples qui , nous l'espèrons , auront éclairé quelque peu votre lanterne , nous allons vous exposer **tous les codages de variables en mémoire** ...

En effet , ce problème apparaît comme très peu connu par les utilisateurs du X-07 et pourtant , il apporte un réel complément à la programmation structurée ...

CODAGE DES VARIABLES

Le codage des variables et des tableaux constitue un passage important à la bonne compréhension des mécanismes internes du X-07 . En effet , une bonne partie des informations que vous désirez conserver lors de votre programmation est conservée dans des variables ...

Il est donc primordial de bien disséquer ce codage qui , d'ailleurs , n'est pas très évident au premier abord ...

2.1 LE CODAGE DES VARIABLES .

Le codage des variables peut être appréhendé grâce à **la fonction VARPTR** du BASIC . Elle affiche l'adresse d'une variable donnée et permet de remonter le codage de cette dernière ...

Afin de bien vous faire comprendre le codage des variables dans la mémoire du CANON , nous avons voulu faire passer un message graphique peuplé d'exemples ... Reportez-vous à la figure 1 .

Voici quelques indications **sur la place occupée** par les divers types de variables que l'on peut rencontrer :

Variable entière (%) : **5 octets** .
Variable en simple précision (!) : **7 octets** .
Variable en double précision (#) : **11 octets** .
Chaîne de caractères ($) : **6 octets + la longueur de la chaîne** .

2.2 LE CODAGE DES TABLEAUX .

Les tableaux de variables autorisent de nombreuses applications très pratiques et surtout très puissantes . La figure 2 expose les différents codages possibles .

Voici quelques indications **sur la taille en octets** qu'ils nécessitent :

* Longueur d'un tableau d'entiers = **10 + 2 * indice maximum**
(une dimension)

* Longueur d'un tableau de nombres en Simple Précision :
12 + 4 * indice maximum
(une dimension)

* Longueur d'un tableau d'entiers :
10 + 2 * (indice 1 maximum + 1) * (indice 2 maximum + 1)
(deux dimensions)

Quelques exemples d'application :

DIM T%(20) --> Nombre d'octets occupés : 50
DIM T!(20) --> Nombre d'octets occupés : 92
DIM T#(20) --> Nombre d'octets occupés : 176

DIM T%(4,5) --> Nombre d'octets occupés : 70
DIM T#(4,5) --> Nombre d'octets occupés : 250

DIM T%(2,4,5) --> Nombre d'octets occupés : 192
DIM T#(2,4,5) --> Nombre d'octets occupés : 732

Nous allons bientôt aborder un sujet passionnant : l'utilisation de certaines routines de la ROM . Alors , accrochez votre ceinture à votre fauteuil car nous allons décoller dans quelques pages !!

CONTENU DE LA MEMOIRE	EXEMPLES D' APPLICATION
Type (2) -3 NOM 1 -2 NOM 2 -1 Octet - significatif <-- VARPTR Octet + significatif **ENTIERS**	En prenant la variable A% = 1234 , on a : 2 41 (A) 00 D2 04
Type (4) -3 NOM 1 -2 NOM 2 -1 Exposant <-- VARPTR 1er chiffre | | | dernier **S.P.**	En prenant CD! = 1234 = 0,1234 * 10^4 : 4 43 (C) 44 (D) 44 — Exposant + 40h = 44h 12 34 00
Type (8) -3 NOM 1 -2 NOM 2 -1 Exposant <-- VARPTR 1er chiffre | | | | | | dernier **DOUBLE PRECISION**	Prenons PI# = 3,1415926535898 : 8 50 (P) 49 (I) 41 — Exposant + 40h = 41h 31 41 59 26 53 58 98
Type (3) -3 NOM 1 -2 NOM 2 -1 Longueur <-- VARPTR Adresse LSB Adresse MSB **CHAINE**	Prenons la variable T$ = "TOTO" : 3 54 (T) 0 4 5B 05 **N.B : les valeurs données dans les cases sont en hexadécimal .**

FIGURE 1 : CODAGE DES VARIABLES

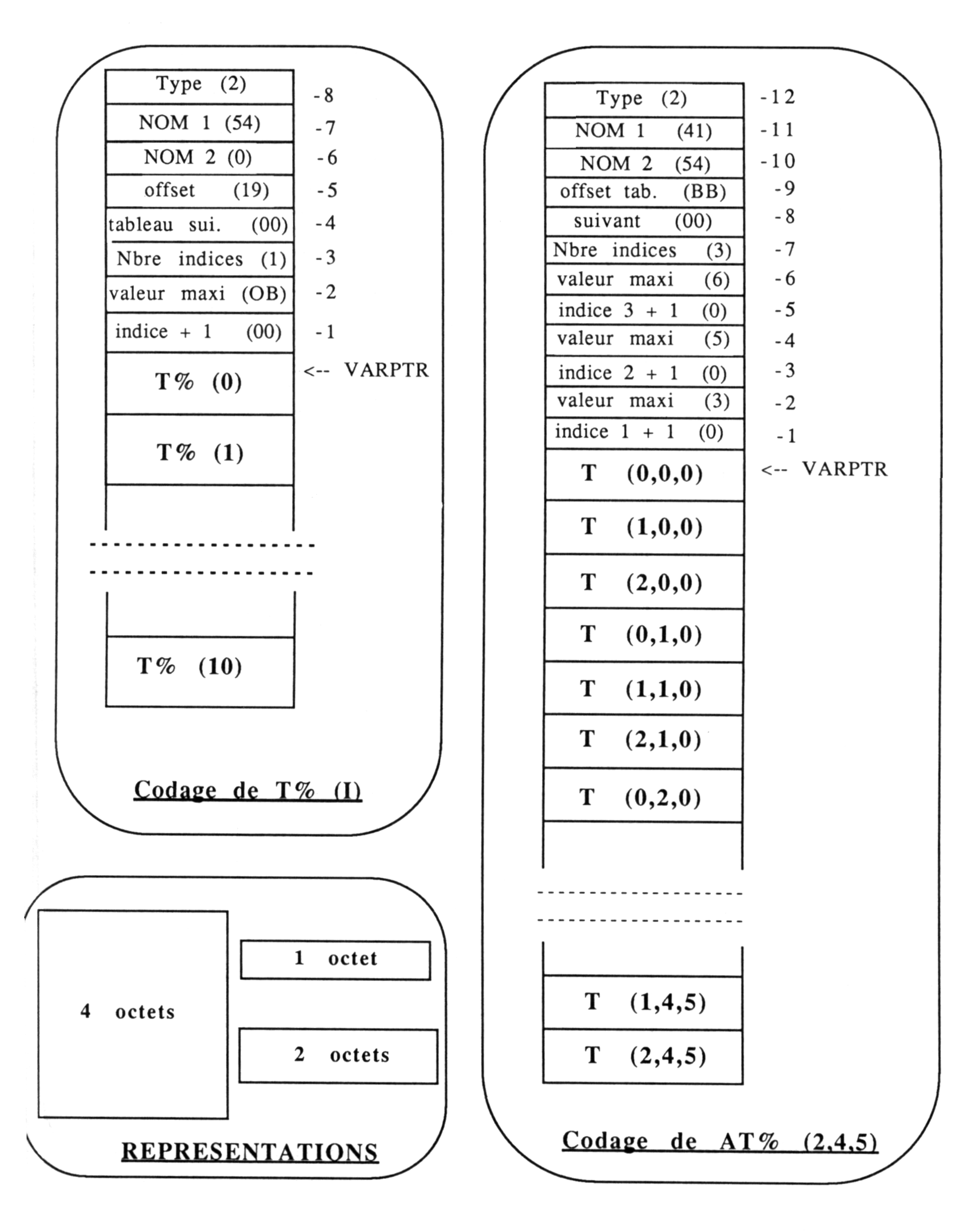


FIGURE 2 : CODAGE DES TABLEAUX

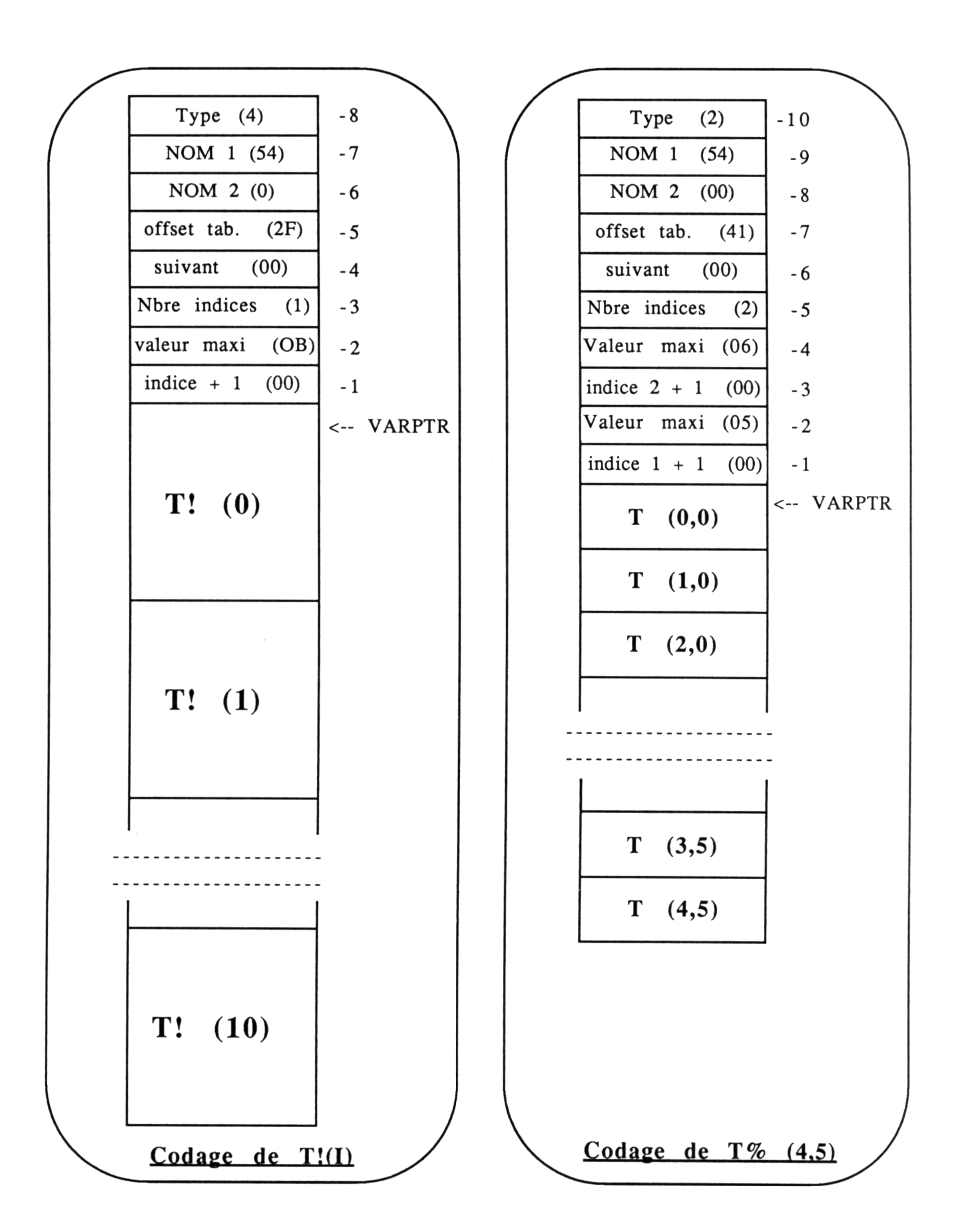


FIGURE 3 : CODAGE DES TABLEAUX

UTILISATION DE ROUTINES DE LA ROM

Ce chapitre est entièrement consacré à **l'utilisation de certaines routines de la mémoire morte du X-07** . En effet , il est très agréable de posséder une pléiade d'adresses diverses , encore faut-il pouvoir ou savoir les utiliser à l'optimum de leurs possibilités ...

C'est en pensant à cet impératif que nous vous avons écrit les quelques routines qui vont suivre . Elles sont très simples à comprendre et vous seront d'une grande utilité dans tous vos logiciels écrits en langage machine .

Evidemment , les programmeurs chevronnés trouveront probablement ces petites routines très classiques ... Nous les invitons donc à passer au chapitre suivant sans autre forme de procès !

Par contre , ceux qui désirent approfondir et optimiser leur programmation vont se délecter ... Plusieurs routines de la ROM du CANON ont été utilisées : pour ceux qui ne les connaissent pas , reportez-vous aux "MYSTERES du X-07" .

D'autre part , les programmes sont présentés sous leur forme "brute" . A vous de les implanter où bon vous semble !! Trois champs sont présents pour chaque listing : LABEL , MNEMONIQUE , COMMENTAIRE .

Entrons sans plus attendre dans le merveilleux monde de la programmation pure...

3.1 LA FONCTION "INKEY$" RETROUVEE ...

L'instruction **"INKEY$"** se montre indispensable en BASIC . Pourquoi ne pas la créer en langage machine ? ... Elle deviendra très vite primordiale : saisie d'options , menu , jeu d'arcade , attente ...

Nous allons pour cela , utiliser une routine très pratique située en **&HC9OA** . Rappelons qu'elle permet de saisir un octet au clavier .

Voici un listing de ce petit sous-programme :

```
INKEY   XOR A            ; Attente de la pression d'une touche
        CALL  $C90A      ; Appel de la routine
        JR Z , INKEY     ; Saut si aucune touche n'est pressée
        RET              ; Retour et le registre A contient le code
                           ASCII de la touche pressée
```

3.2 TRACE DE DROITES .

La routine exposée va vous permettre de **tracer un nombre donné de droites sur l'écran LCD du X-07** .

Le principe est simple . On utilise **le sous-processeur T6834** en lui envoyant les coordonnées de la droite à tracer (4 données indispensables par droite : Xi , Yi , Xf , Yf) . Ces nombres sont contenus dans un espace repéré par le registre HL . Le T6834 trace alors le segment désiré grâce à sa routine N°14 (en hexadécimal) .

Dès qu'il a terminé , il recommence si le nombre de droites (contenu dans le registre B) n'est pas nul ...

Donc , avant d'utiliser la routine , il faut :

_ Remplir une zone tampon avec vos coordonées .
_ Charger **le registre HL** avec l'adresse de début de cette zone .
_ Charger **le registre B** avec le nombre de droites à tracer .

Voici le listing :

```
DROITE  PUSH BC          ; Sauvegarde de BC (nombre de droites)
        LD A , $14       ; Chargement de A avec la commande 14h
        LD BC , $400     ; B contient le chiffre 4 (4 paramètres) et
                           C contient 0 car il n'y a pas d'octet de
                           réponse attendu
        CALL  $C92F      ; Appel du sous-processeur
        POP BC           ; On récupère le nombre de droites ...
        DJNZ DROITE      ; ... afin de vérifier si l'on a terminé ...
        RET              ; ... si oui , on quitte la routine
```

Remarquons la rapidité de tracé qui n'est pas désagréable . De plus , la routine $C92F incrémente automatiquement le registre HL ... Sublime , non ?!!

3.3 IMPLANTATION DE CARACTERES GRAPHIQUES .

Cette routine va vous permettre de définir **des caractères affichables pour les codes ASCII 80 à 9F et E0 à FF** .

L'intérêt est de pouvoir définir un nombre de caractères en une seule fois dès que la table est entrée en mémoire .

Le principe est le suivant ... Une table des codes est repérée par le registre HL . Cette table doit être structurée de la manière suivante : code du caractère ASCII , 8 codes , code du caractère ASCII , 8 codes , etc ...

Le registre B doit être préalablement chargé avec le nombre de caractères à implanter .

La routine utilisée (commande **N°1A** du sous-processeur) se charge d'implanter les codes aux endroits nécessaires ...

Voici le listing :

```
CRECAR  LD B , nbre de car.   ; Chargement de B avec le nombre de
                                caractères à implanter
        LD HL , TCAR          ; Chargement de HL avec l'adresse de
                                début de la table des codes
CC      PUSH BC               ; Sauvegarde du nombre de caractères
        LD A , $1A            ; Chargement de A avec le N° de la com-
                                mande
        LD BC , $900          ; B contient le nombre de paramètres (9)
                                et C le nombre d'octets attendus (0)
        CALL $C92F            ; Appel du sous-processeur
        POP BC                ; On récupère le nombre de caractères ...
        DJNZ CC               ; ... afin de vérifier si l'on a terminé ...
        RET                   ; ... si oui , on quitte le sous-programme
```

3.4 COMMENT TESTER LES CURSEURS ? ...

Tester les curseurs est une chose fréquente quand l'on programme en BASIC . Afin de pouvoir les utiliser aussi sous ASSEMBLEUR , nous vous avons préparé une petite routine très fiable ...

Le principe est encore une fois de passer par le T6834 (Nous l'avons pris en affection depuis que nous le connaissons si bien ...) grâce à la routine $C92F.

Notons que les registres A , HL et BC sont modifiés .

Voici le listing :

```
FLECHE  CALL $COBD      ; Les tampons clavier et sous-processeur
                          sont vidés
        LD A , $82      ; Commande N°2 du T6834 (STICK)
        LD BC , $1      ; Un paramètre est attendu en retour
        PUSH DE         ; Sauvegarde de DE (si nécessaire ...)
        LD DE , BUF     ; Le registre DE est chargé avec l'adresse
                          d'un tampon utilisateur que vous devez
                          définir . Ce tampon sert à sauvegarder la
                          réponse du T6834
        CALL $C92F      ; Appel du T6834
        LD A , (DE)     ; On charge A avec la réponse contenue
                          dans le BUFfer utilisateur
        POP DE          ; On récupère DE
        CP $33          ; Curseur droit pressé ?
        JR Z , DROITE   ; si oui , saut à une routine de traitement
        CP $37          ; Curseur gauche pressé ?
        JR Z , GAUCHE   ; si oui , saut ...
        CP $35          ; Curseur bas pressé ?
        JR Z , BAS      ; si oui , saut ...
        CP $31          ; Curseur haut pressé ?
        JR Z , HAUT     ; si oui , saut ...
        RET             ; Le registre A contient la valeur $30 si
                          aucun curseur n'a été pressé ...
```

3.5 TESTER UNE TOUCHE PARTICULIERE ...

Cette petite routine va vous exposer **la manière de tester une touche quelconque du clavier** ...

Le sous-processeur est encore mis à contribution grâce à sa routine N°28 (en hexadécimal , toujours ...) . Les registres **A , HL , BC et DE** sont modifiés après l'emploi de cette routine : n'oubliez pas de les sauvegarder si nécessaire !

N'oubliez pas de charger le registre A avec la touche que vous désirez tester avant de lancer le sous-programme !

Après que la routine se soit exécutée , une touche aura été pressée si le **drapeau Z du registre A est à 1** . Sinon , il sera normalement à 0 ...

Voici le listing :

```
TOUCHE  LD HL , BUF + 4      ; Le registre HL est chargé avec l'adresse
                               du buffer utilisateur augmenté de 4
        LD (HL) , A          ; Le contenu de A (code de la touche à
                               tester) est stocké dans le buffer
        LD A , $28           ; A est chargé avec la commande N°28
        LD BC , $101         ; B contient le nombre de paramètres (1)
                               et C le nombre d'octets attendus (1)
        LD DE , BUF + 2      ; Le registre DE est chargé avec l'adresse
                               du buffer augmenté de 2 unités
        CALL $C92F           ; Appel du T6834
        LD A , (DE)          ; A est chargé avec la réponse du T6834
                               stockée dans le buffer . Si A = 0 , la
                               touche a été pressée sinon A = $FF
        OR A                 ; Les indicateurs Z et S "ressortent" ...
        RET                  ; Retour
```

3.6 COMMENT CREER UNE TEMPORISATION ? ...

Enfin **une routine permettant la temporisation** ... C'est normal que tout le monde aime ce genre de programmes : ce sont les seuls permettant de faire une sieste bien méritée !!

Le principe est tellement simple qu'il n'y a rien à expliquer !! Les instructions utilisées sont spécifiques au Z-80 en général donc pas de problème pour les comprendre ...

Le registre BC est chargé avec la durée de temporisation désirée ... Bon somme !!

Voici le listing :

```
DELAY   DEC B                ; Décrémentation du registre B
        LD A , B             ; Chargement de A avec B pour effectuer
                               une comparaison
        OR C                 ; Test ...
        JR NZ , DELAY        ; S'il fait encore nuit , on continue à tempo-
                               riser !!
        RET                  ; Le jour se lève : fini de dormir !
```

3.7 "MUSIQUE , MAESTRO !!"

Vous allez enfin pouvoir réaliser votre vieux rêve : **réécrire la 9ème symphonie de Beethoven sur CANON X-07** ... Epoustouflant !!

La routine **$C2DF** est utilisée pour créer toute sorte de sons : elle est un peu compliquée à utiliser mais les résultats en valent la peine !

Tous les registres sont modifiés . Un seul registre à charger : **IX** . Il doit impérativement contenir l'adresse de votre mélodie (codée sous formes de notes , bien entendu , dans une zone mémoire) . Le tampon est structuré sous cette forme : note , durée , note , durée ... Dès que **la valeur 255 est trouvée**, le X-07 sait qu'il n'y a plus de note ensuite ...

Le saxophone est mort ... Vive le X-07 !! (Enfin ... presque !!)

Voici le listing :

```
MUSIC   LD A , (IX + $0)      ; A est chargé avec l'octet pointé par IX
        CP $FF                ; Dernière note ?
        RET Z                 ; Si oui , on quitte la symphonie ...
        LD E , A              ; Le registre E contient la note
        LD B , (IX + $1)      ; Le registre B contient la durée
        LD D , 0              ; Le registre D est nul
        LD C , D              ; Le registre C est nul (un octet de gagné !)
        INC IX                ; Incrémentation double du registre IX
        INC IX                ; afin de parvenir à la note suivante
        LD HL , RTMU          ; On va empiler l'adresse de retour pour la
        PUSH HL               ; routine $C2DF (caractéristique ...)
        PUSH BC               ; Sauvegarde de BC
        PUSH DE               ; Sauvegarde de DE
        POP IY                ; IY contient la durée
        POP HL                ; HL contient la note
        CALL $C2DF            ; Appel de la routine musicale
RTMU    LD HL , ($0E)         ; Compteur de note
        LD A , H              ; Test du registre HL : est-il bien égal à 0 ?
        OR L                  ; suite du test ...
        JR NZ , RTMU          ; On "boucle" si la note est en cours ...
        JR MUSIC              ; ... Sinon , on passe à la note suivante ...
```

Mozart en herbe , c'est le moment de prouver vos talents !! Au début de la page suivante , un bel exemple vous est donné ...

Exemple de mélodie :

```
AIR1     DB   17 , 4 , 17 , 4 , 17 , 4 , 13 , 16 , 0 , 16 , 15 , 4
         DB   15 , 4 , 15 , 4 , 12 , 16 , 0 , 16 , 17 , 4 , 17 , 4
         DB   17 , 4 , 17 , 4 , 13 , 4 ,18 , 4 , 18 , 4 , 18 , 4
         DB   17 , 4 , 25 , 4 , 25 , 4 , 25 , 4 , 22 , 16 , 255
```

Remarques : si la note est égale à **0** , un bruit est généré . D'autre part , n' oubliez pas le "terminateur" (**255**) .

3.8 FONCTIONS SET , RESET , POINT .

Ces fonctions indispensables au "graphiste canoniste" sont très faciles à obtenir via le sous-processeur que vous devez commencer à bien connaître ... Après toutes les manipulations faites avec lui , il faut bien reconnaître qu'il facilite beaucoup la vie du programmeur ... En effet , vous n'avez guère besoin de connaître toutes les méthodes de travail de l'ensemble des routines !

Il suffit donc d'envoyer le numéro de la commande désirée (**Soit $11 pour SET , $13 pour POINT et $12 pour RESET**) au T6834 grâce à notre inaltérable routine $C92F . Avant d'utiliser la routine , choisissez une zone buffer et stockez-y les deux coordonnées du PIXEL (X et Y) sur lequel vous désirez travailler ...

Voici le listing :

```
SET      LD A , $11         ; Chargement de A avec la commande $11
         JR SET1            ; Saut à la routine de traitement
POINT    LD A , $13         ; Chargement de A avec la commande $13
         JR SET1            ; Saut ...
RESET    LD A , $12         ; Chargement de A avec la commande $12
SET1     LD HL , BUF        ; Chargement de HL avec l'adresse du
                              buffer utilisateur
         LD BC , $200       ; B contient le nombre de paramètres
                              envoyés (2) et C le nombre d'octets
                              attendus (en l'occurence , 0)
         JP $C92F           ; Le traitement graphique se fait ...
```

3.9 ET SI L' ON PARLAIT D' INPUT ?

Un logiciel informatique a obligatoirement besoin de données pour travailler ... Une grande partie de ces données sont encore entrées au clavier et la fonction "INPUT" joue un rôle déterminant dans ces stockages ... Nous vous proposons donc de la recréer !

En effet , la routine va créer la fonction " **INPUT "message ";A$** " d'une manière très simple . Quelques fonctions de la ROM sont utilisées telles que **$FEF7 et $EBF2** : l'une affiche un message et l'autre permet d'obtenir l'entrée clavier . Notons que le message ne doit pas dépasser 20 caractères dans le cas présent .

Voici le listing :

```
ACQMES  LD HL , MESSAGE     ; HL contient l'adresse du message qui doit
                              être impérativement terminé par un 0 .
        CALL $FEF7          ; Le message est affiché ...
        CALL $EBF2          ; L'entrée clavier est obtenue .
        INC HL              ; On incrémente HL pour obtenir le début
                              du tampon d'entrée étant donné que la
                              routine $EBF2 fait pointer HL sur cette
                              adresse - 1 .
        LD DE , BUF         ; DE est chargé avec l'adresse du tampon
                              utilisateur
        LD C , 0            ; C = compteur
ACQ     LD A , (HL)         ; Transfert vers tampon interne
        LD (DE) , A         ; suite du transfert
        OR A                ; Test du 0 situé en fin de "A$"
        JR Z , ACQ2         ; Saut à ACQ2 si réalisé ...
        INC HL              ; caractère suivant
        INC DE              ;        "
        INC C               ; incrémentation de C
        LD A , C            ; Longueur maximale atteinte ?
        CP 20               ; Test sur la longueur (ici 20)
        JR C , ACQ          ; Si la longueur est atteinte , on sort de la
                              boucle ...
ACQ2    LD A , C            ;
        CP 0                ; Si la chaîne est vide , on recommence ...
        JR Z , ACQMES       ;
        LD (LONG) , A       ; Sauvegarde de la longueur
        RET                 ; Retour
```

3.10 ENTREE D' UN NOMBRE .

Nous allons continuer sur notre lancée en vous détaillant une routine équivalente ... Cette fois , par contre , vous pourrez entrer uniquement des nombres entiers . La fonction correspond donc à " **INPUT "message ";A%** " .

Notons l'utilisation de routines permettant d'économiser un certain nombre d'octets ... En l'occurence , **les routines RST $10 et $F595** permettent quantité d'applications : nous vous conseillons de bien les étudier dans "Les MYSTERES du X-07" car elles vous permettront de gagner beaucoup de place en mémoire !

Voici le listing :

```
ACQNUM  LD HL , MESSAGE      ;
        CALL $FEF7           ; Affichage du message et entrée clavier
        CALL $EBF2           ;
        RST $10              ; Teste si un nombre a bien été entré ...
        CALL $F595           ; Transfère le nombre dans le registre DE
```

3.11 UTILISATION DE LA X-710 .

En envoyant **une chaîne de codes déterminés** à l'imprimante graphique , vous pouvez la contrôler aisément à partir de l'ASSEMBLEUR et lui faire réaliser de très belles choses ...

La routine **$CEF7** émet le contenu du registre A vers la X-710 . La chaîne de codes doit être , comme à l'accoutumé , stockée dans un buffer .

Voici le listing :

```
        LD HL , CHAINE       ; HL est chargé avec l'adresse du début de
                               la chaîne
LPRINT  LD A , (HL)          ; A est chargé avec l'un des codes de la
                               chaîne
        OR A                 ; Test afin de savoir si le dernier code a été
        RET Z                ; envoyé vers la X-710
        PUSH HL              ; Sauvegarde de l'adresse des codes
        CALL $CEF7           ; Transmission du code contenu dans A
        POP HL               ; On récupère l'adresse du prochain code
        INC HL               ; Incrémentation du registre HL
        JR LPRINT            ; On continue le cycle ...
```

Voici un exemple de chaîne à transmettre à la X-710 :

18 , 83 , 48 , 13 , 67 , 49 , 77 , 48 , 45 , 54 , 13 , 65 , 13 , 10 , 00

18 : mode graphique
83 , 48 : taille "S0"
13 : retour chariot
67 , 49 : couleur "C1"
77 , 48 , 45 , 54 : commande "M 0,-6" --> Remontée de 6 points
13 : retour chariot
65 : lettre A (mode texte)
13 : retour chariot
10 : commande Line Feed (déroulement du papier)
00 : fin des commandes

Ce chapitre se termine donc sur cette utilisation de notre traceur national, en l'occurence , la X-710 .

Nous osons espérer que nous avons été assez clair ... Si c'est bien le cas , vous allez bientôt pouvoir vous précipiter sur les multiples applications situées en fin d'ouvrage .

Mais auparavant , intéressons-nous à un certain crochet d'erreur situé quelque part dans la zone système ... c'est fou ce que l'on peut faire avec !!

EXTENSION DE FONCTIONS

4.1 INTRODUCTION .

Il existe sur le X-07 plusieurs crochets ("hoocks") permettant de prendre le contrôle du CANON dans certaines situations . **On peut ainsi modifier les réactions de la machine face à une erreur particulière** .

Pour illustrer ce phénomène , nous allons écrire deux routines dérivant ce crochet d'erreur : la première permettra d'obtenir un message d'erreur en français et la deuxième rajoutera une instruction BASIC supplémentaire au X-07 . Notons au passage que le CLUB C7 a abondamment parlé de ce crochet dans ses gazettes 2 et 5 .

4.1.1 LE PRINCIPE .

Lorsque le CANON rencontre une erreur , il exécute un sous-programme spécial ... Mais avant cela , il exécute la routine située à **l'adresse ABh** . Normalement , on trouve l'instruction RET (code C9h) à cette adresse ... Rien ne nous empêche de placer un saut à cette adresse afin de détourner ce crochet sur l'une de nos petites routines !

Notons que notre tâche se trouve simplifiée car le numéro de l'erreur se trouve dans **le registre E** du microprocesseur .

4.1.2 LES MESSAGES D'ERREUR EN FRANCAIS ...

Nous allons donc en premier lieu écrire une routine permettant d'afficher les messages d'erreur en français . Afin de simplifier le programme , nous nous limiterons aux trois erreurs les plus courantes :

_ Erreur de syntaxe : code 2 .
_ Erreur de débordement : code 6 .
_ Erreur de numéro de ligne indéfini : code 8 .

Le travail à réaliser en est d'autant simplifié car traiter tous les cas d'erreur possibles aurait demander une table assez grande contenant tous les types d'erreurs . Ici , la routine exposée est relativement simple ...

4.2 LE DETOURNEMENT DU CROCHET .

Pour détourner ce crochet , **on implante à la place du RET situé en ABh un JUMP XXXX** correspondant au début de notre routine personnelle de traitement . Après avoir implanté cette "dérivation" , on teste au début du logiciel si le type d'erreur qui survient correspond bien aux erreurs que l'on désire traduire en français ... Si c'est effectivement le cas , on continue à exécuter notre routine sinon on revient au crochet (grâce à un RET) et le X-07 affiche l'erreur initiale .

Voici le listing de ce programme :

```
5 ' [                               ; Messages d'erreur en français
10 ' ORG $1C00                      ; Les lignes 10 à 60 représentent la
20 ' LD A.$C3                       ; modification du crochet d'erreur
30 ' LD ($AB).A                     ; situé à l'adresse $AB .
40 ' LD HL.#DB                      ; EXEC &H1C00 mettra en place la
50 ' LD ($AC).HL                    ; nouvelle adresse de traitement des
60 ' RET                            ; erreurs .
70 ' #DB LD A.E                     ; Le type d'erreur est chargé dans A
80 ' CP $2                          ; Si l'erreur qui survient est syntaxi-
90 ' JR Z.#E3                       ; que , on saute à #E3 .
100 ' CP $6                         ; Si l'erreur qui survient est de type
110 ' JR Z.#E6                      ; "débordement" , on saute à #E6 .
120 ' CP $8                         ; Si l'erreur qui survient est de type
130 ' JR Z.#E8                      ; numéro 8 , on saute à #E8 .
140 ' RET                           ; Autre erreur : retour ...
150 ' #E3 LD HL.#SY                 ; Les lignes 150 à 180 constituent le
160 ' CALL $FEF7                    ; sous-programme de traitement des
170 ' CALL #NO                      ; erreurs de syntaxe .
180 ' JP $F23D                      ; Saut à l'attente-curseur .
190 ' #E6 LD HL.#DE                 ; Les lignes 190 à 220 constituent le
200 ' CALL $FEF7                    ; sous-programme de traitement des
210 ' CALL #NO                      ; erreurs de débordement .
220 ' JP $F23D                      ; Saut à l'attente-curseur .
230 ' #E8 LD HL.#ID                 ; Les lignes 230 à 260 constituent le
240 ' CALL $FEF7                    ; sous-programme de traitement des
250 ' CALL #NO                      ; erreurs numéro 8 .
260 ' JP $F23D                      ; Saut à l'attente-curseur .
270 ' #SY DEFM erreur de syntaxe    ; Message d'erreur
280 ' DEFB $0A , 0D , 00
```

```
290 ' #DE DEFM débordement en      ; Message d'erreur
300 ' DEFB $0A , 0D , 00
310 ' #ID DEFM ligne indéfinie     ; Message d'erreur
320 ' DEFB $0A , 0D , 00
330 ' #NO LD HL.#N1                ; Affichage du numéro de ligne
340 ' CALL $FEF7
350 ' LD HL.($01DB)
360 ' CALL $BB98
370 ' RET
380 '#N1 DEFM LIGNE
390 ' DEFB $20,00
400 ' ]                            ; Fin du programme
```

Afin d'obtenir une présentation plus claire , on fait passer le X-07 à la ligne après l'affichage de chaque type d'erreur et ceci en envoyant les caractères **0A et 0D** . L'adresse **1DBh** contient le numéro de la dernière ligne exécutée .

Notons que la routine **$F23D** place le X-07 en mode "attente-curseur" ... Ceci a pour effet d'arrêter le programme pour ne pas retourner au traitement d'erreur originel .

Voici pour terminer cet exemple , le listing BASIC de cette routine :

```
10 REM Messages d'erreurs en français
20 FOR I = &H1C00 to &H1C00 + 140
30 READ A$ : POKE I , vAL ("&H"+A$)
40 NEXT
50 PRINT "EXEC &H1C00 sera nécessaire au détournement des futures
     erreurs" .
60 DATA 3E , C3 , 32 , AB , 0 , 21 , C , 1C , 22 , AC , 0 , C9 , 7B , 32 , 0 , 1B ,
     FE , 2 , 28 , 9 , FE , 6 , 28 , 11 , FE , 8
70 DATA 28 , 19 , C9 , 21 , 41 , 1C , CD , F7 , FE , CD , 79 , 1C , C3 , 3D , F2 ,
     21 , 56 , 1C , CD , F7 , FE , CD , 79 , 1C
80 DATA C3 , 3D , F2 , 21 , 67 , 1C , CD , F7 , FE , CD , 79 , 1C , C3 , 3D , F2 ,
     45 , 52 , 52 , 45 , 55 , 52 , 45 , 20
90 DATA 44 , 45 , 20 , 53 , 59 , 4D , 54 , 41 , 58 , 45 , A , D , 0 , 44 , 45 ,
     42 , 4F , 52 , 4 , 45 , 4D , 45 , 4E , 54 , 20
100 DATA 45 , 4E , A , D , 0 , 4C , 49 , 47 , 4E , 45 , 20 , 49 , 4E , 44 , 45 ,
     46 , 49 , 4E , 49 , 45 , A , D , 0 , 21 , 86
110 DATA 1C , CD , F7 , FE , 2A , DB , 1 , CD , 98 , BB , C9 , 4C , 49 , 47 , 4E ,
     45 , 20 , 0
```

4.3 UNE INSTRUCTION SUPPLEMENTAIRE ...

4.3.1 L' INSTRUCTION "CARTE" .

Nous allons maintenant mettre en place une instruction supplémentaire , la commande "CARTE" . Sa fonction sera d'engendrer un "Create system ?" permettant de vider la carte .

Le principe de la routine est de détourner le drapeau d'erreur puis de vérifier que l'erreur survenue est bien une erreur de syntaxe . Ensuite , il faut vérifier que cette erreur a bien été engendrée par le mot "CARTE" . Si ce n'est pas le cas , on retournera au traitement normal d'erreur .

Comment tester lettre par lettre le mot "CARTE" ? Grâce au registre HL qui pointe sur le caractère suivant l'erreur .

Voici le listing de la routine :

```
5 ' [                    ; Début de la routine .
10 ' LD A.$C3            ; Les lignes 10 à 50 détournent le
20 ' LD ($AB).A          ; crochet d'erreur .
30 ' LD HL.#DB
40 ' LD ($AC).HL
50 ' RET
60 ' #DB LD A.E          ; Si ce n'est pas une erreur de syntaxe
70 ' CP $2               ; on continue le traitement normal .
80 ' RET NZ
90 ' CALL #DE            ; Les lignes 90 à 230 vérifient la
100 ' CP "E"             ; justesse du mot "CARTE" ...
110 ' RET NZ
120 ' CALL #DE
130 ' CP "T"
140 ' RET NZ
150 ' CALL #DE
160 ' CP "R"
170 ' RET NZ
180 ' CALL #DE
190 ' CP "A"
200 ' RET NZ
210 ' CALL #DE
220 ' CP "C"
230 ' RET NZ
240 ' JP $C4D1           ; On saute à la routine de "Create ..."
250 #DE DEC HL
260 ' LD A.(HL)          ; HL pointe sur le mot qui a entrainé
270 ' RET                ; l'erreur ...
280 ' ]
```

Voici le listing BASIC de cette petite routine :

```
10 ' Instruction CARTE
20 FOR I = &H1C00 TO &H1C00 + 51
30 READ A$ : POKE I , VAL ("&H" + A$)
40 NEXT
50 EXEC &H1C00
60 DATA 3E , C3 , 32 , AB , 0 , 21 , C , 1C , 22 , AC , 0 , C9 , 7B , FE , 2 , C0 ,
   CD , 31 , 1C , FE , 45 , C0 , CD , 31 , 1C , FE , 54 , C0 , CD , 31 , 1C , FE ,
   52 , C0 , CD , 31 , 1C , FE , 41 , C0 , CD , 31 , 1C , FE , 43 , C0 , C3 , D1 ,
   C4 , 2B , 7E , C9
```

4.3.2 L'INSTRUCTION "PAINT" DETOURNEE ...

Pour terminer ce chapitre , voici une petite routine rajoutant l'instruction "PAINT" au BASIC du X-07 . Cette commande aura uniquement pour fonction d'afficher le message "PAINT" à l'écran .

Le drapeau de "PAINT" (instruction utilisée en vidéo avec la X-720) se situe en **99h** . Nous allons procéder de la même manière que pour le détournement des erreurs en plaçant à cette adresse un **JP XXXX** nous envoyant à l'adresse de notre routine ...

Voici le listing de ce programme :

```
5 ' [                        ; Début du programme .
10 ' ORG $1C00               ; Les lignes 10 à 70 détournent le
30 ' LD A.$C3                ; crochet d'erreur sur notre routine .
40 ' LD ($99).A
50 ' LD HL.#DB
60 ' LD ($9A).HL
70 ' RET
80 ' #DB PUSH HL             ; Les lignes 80 à 110 affichent le
90 ' LD HL.#ME               ; message ...
100 ' CALL $D5B0
110 ' POP HL
120 ' RET                    ; On continue le programme ...
130 ' #ME DEFM PAINT
140 ' DEFB $00
150 ' ]
```

Les applications de ces crochets d'erreur sont multiples comme vous avez pu le constater ... Nous allons maintenant nous plonger dans la partie "HARD" de cet ouvrage ...

2ème PARTIE :

... Où l'on continue par du HARD ...

ENTREES/SORTIES & INTERRUPTIONS

Ce chapitre sur **les entrées/sorties et les interruptions** constitue la quintessence du CANON ... En effet , le X-07 est très réputé pour la multiplicité de ses entrées/sorties et nous avons pensé nous aventurer un peu sur ce terrain relativement inexploré ...

Notons que le CLUB C7 a édité un très bon dossier (**"Le X-07 s'évade"**) sur le sujet : notre machine adorée se branche alors sur une lampe , un réveil , un magnétophone... et les commande remarquablement !! Avis aux amateurs aimant les sensations fortes !!

5.1 LES INTERRUPTIONS .

5.1.1 GENERALITES .

Les interruptions constituent des méthodes d'entrées/sorties un peu particulières . En effet , les entrées/sorties programmées accusent deux inconvénients majeurs :

_ Elles font perdre du temps au microprocesseur étant donné qu'il doit examiner l'état de tous les périphériques .

_ Après avoir déterminé le périphérique devant entrer en contact avec lui , le Z-80 exécute le programme correspondant . Malheureusement , ce temps de réponse peut devenir critique si un périphérique véloce a besoin du processeur ... C'est le cas ici avec l'écran ou les entrées/sorties sérielles .

Une solution a été découverte en reliant chaque périphérique à **une ligne d'interruption** . Le Z-80 examine à la fin de chaque cycle-machine (instruction) ces lignes d'interruption . Si un arrêt est demandé , le microprocesseur **sauvegarde l'adresse à laquelle il travaillait** afin d'exécuter la routine correspondant à l'interruption demandée .

De plus , quand une interruption est en cours , il ne faut peut être pas qu'une autre interruption l'interrompe ... Pour éviter cela , il est possible d'interdire les interruptions en leur donnant **un niveau de priorité différent** .

A ce moment là , l'interruption possèdant le plus haut degré de priorité ne pourra être stoppée par aucune autre mais , par contre , cette interruption pourra interrompre toutes les autres !!

5.1.2 LES TYPES D' INTERRUPTION .

Le CANON X-07 est doté de **deux types** distincts d'interruptions :

_ **Les interruptions A , B , C** ayant chacune un niveau défini de responsabilités .

* INT A : interruption provenant du sous-processeur T6834 .
  Le programme d'interruption correspondant se situe en **C799h** .
* INT B : interruption sérielle . Un saut est provoqué à l'adresse **C7A3h** .
* INT C : cette interruption n'est utilisée que quand le X-07 est relié à la X-720 . Une horloge reliée à cette ligne d'interruption oblige le CANON à afficher le curseur .

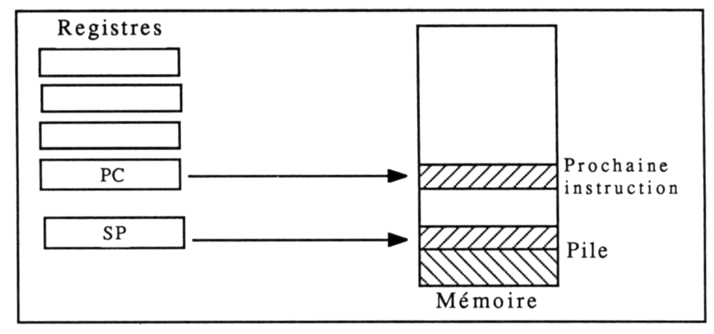
_ L'interruption **NMI** n'est pas utilisée sur le X-07 . C'est l'interruption possédant le plus haut niveau de priorité ... En effet , NMI se traduit en anglais par "No Masquable Interrupt" , ce qui signifie bien évidemment "**interruption non masquable"** . Quelque soit la tâche effectuée par le X-07, il suffit qu'un signal d'interruption parvienne sur la ligne NMI pour que le CANON exécute immédiatement le programme correspondant . Notons que cette interruption très particulière va nous être très utile ...

5.1.3 LE TRAITEMENT DES INTERRUPTIONS .

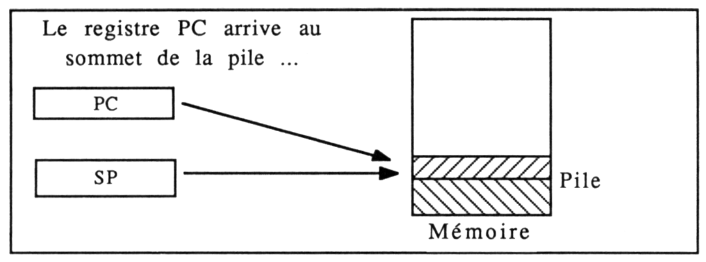
Une fois l'interruption acceptée , il faut impérativement satisfaire "le demandeur" sans perdre le fil du programme en cours . Ceci implique **une sauvegarde des registres dans la pile** , le minimum étant de sauver le compteur ordinal (PC) afin de pouvoir y placer l'adresse de la routine d'interruption à exécuter .

Après avoir effectué le traitement demandé par le périphérique , le microprocesseur peut reprendre tranquillement le fil de son autre programme au point où il l'avait quitté grâce à l'adresse sauvegardée sur la pile (correspondant à PC) . La figure exposée à la page suivante vous permettra de bien assimiler le processus .

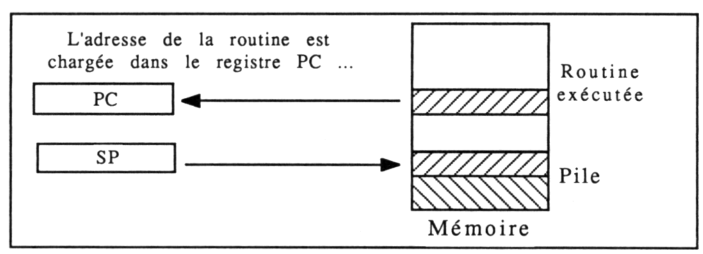
Les interruptions peuvent être déconnectées ou reconnectées à l'aide de deux ordres du NSC 800 : **EI et DI** .



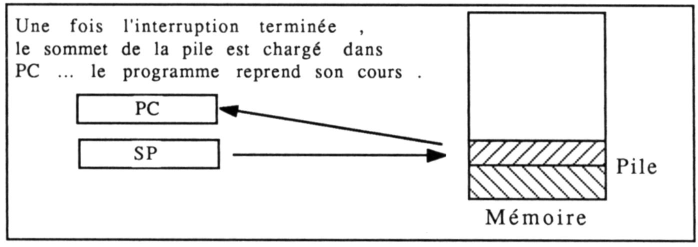
FONCTIONNEMENT NORMAL



ARRIVEE DE L' INTERRUPTION



ROUTINE EXECUTEE



INTERRUPTION TERMINEE

FIGURE 4 : LE TRAITEMENT DES INTERRUPTIONS

En traduisant ces deux abréviations , on obtient Enable Interrupt (interruption autorisée) et Disable Interrupt (interruption interdite) .

En général , on interdit les interruptions lorsque l'on commence le traitement d'une interruption de faible priorité afin de ne pas être dérangé par un autre arrêt . Mais n'oubliez surtout pas de tout remettre en ordre dès que la partie critique de votre routine est terminée grâce à l'instruction EI .

5.1.4 L'UTILISATION DES INTERRUPTIONS SUR LE X-07 .

Le fonctionnement du X-07 est intimement lié à un sous-processeur nommé très poétiquement "T6834" ... Nous en avons abondamment parlé depuis le début de cet ouvrage et nous allons utiliser la ligne d'interruption B afin de détecter les interruptions provenant de ce sous-processeur . Par exemple , un appui sur la touche BREAK ou OFF constitue une interruption .

En fait , les interruptions provoquent des sauts à quatre adresses différentes suivant le type d'arrêt demandé . Ainsi :

* **INT A** provoque un saut en **3Ch** .
* **INT B** provoque un saut en **34h** .
* **INT C** provoque un saut en **2Ch** .
* **NMI** provoque un saut en **66h** .

On peut donc écrire à ces adresses soit le programme que l'on désire faire effectuer au X-07 , soit un "JP XXXX" afin que le Z-80 aille exécuter le programme se trouvant à cette adresse .

Sur le CANON , cette deuxième solution a été pronée et l'on trouve logiquement JP C7A3 , JP C799 , etc ...

Cette solution va nous rendre un grand service étant donné que ces adresses se trouvent en RAM ... Nous allons donc pouvoir les modifier à volonté et découvrir véritablement ce qui s'y passe !

En reprenant l'exemple du T6834 , nous nous rappelons que ce dernier envoie un signal sur l'une des pattes du NSC 800 lorqu'il désire communiquer . Si l'interruption est prise en compte , le microprocesseur ira effectuer la routine siégeant à l'adresse 3Ch . Or , l'adresse 3Ch contient l'instruction de saut JP C799 ... Par conséquent , le programme d'interruption continuera sa course à l'adresse C799h !

De même , si une interruption parvient sur la patte NMI du NSC 800 , ce dernier exécutera les instructions situées à l'adresse 66h ... Mais , notre bon vieux processeur trouvera la directive RET (code C9h) à cet endroit et reprendra l'exécution normale du logiciel étant donné que l'interruption NMI n'est pas utilisée sur le CANON . **Justement , nous allons détourner cette interruption et jour un peu avec les entrées/sorties ...**

5.1.5 APPLICATIONS PRATIQUES .

Les interruptions étant généralement déclenchées par des périphériques, ce sont des pattes spécialisées du microprocesseur qui se chargent de leur détection . Nous allons essayer de les utiliser afin de bien comprendre ce qui se passe lors d'un arrêt demandé .

Pour cela , il nous faut un peu de matériel ... Par exemple , un doigt pour retirer le couvercle du bus (prise située à l'arrière du X-07 : port d'extension) , une paire de lunettes pour lire le brochage (page 127 du guide de l'utilisateur) et un petit morceau de fil électrique pour nous livrer à quelques liaisons coupables !

L'expérience est relativement simple . **Les bornes correspondant aux interruptions A et NMI se trouvent sur le connecteur arrière du X-07** . Après avoir changé les adresses de saut d'une de ces routines , nous provoquerons un saut en reliant la broche correspondante au +5 volts ou au 0 volt , selon la tension requise .

Pour cette expérimentation , nous utiliserons **l'interruption NMI** . En effet , cette borne est sensible au niveau haut (+5 V) alors que INT A est sensible au niveau bas (0 V) , donc plus "sensible" électriquement . Le brochage du connecteur est décrit à la page suivante : la broche NMI correspond à la broche 20 ; le +5 V est présent sur les broches 39 et 40 .

Dès que nous désirerons créer notre interruption , **il nous suffira d'un "bref" contact entre ces deux bornes** .

5.1.5.1 La première expérience .

Nous allons mixer un peu de BASIC et de langage machine uniquement pour une application amusante .

Le but de la routine sera de faire compter le X-07 de 0 à 10000... De plus, dès qu'une interruption se produira , il devra émettre un BEEP sans s'arréter de compter !! Enfin du multitâche sur X-07 !!

En fait , la routine va être composée de deux petits programmes :

_ Le premier va se charger de modifier l'adresse de saut du NMI afin que le X-07 puisse exécuter notre programme dès que nous relierons nos deux bornes ...

_ Le deuxième s'occupera d'émettre un BEEP . Pour cela , l'écriture du caractère "BELL" suffit (code ASCII N°7) .

Le programme écrit en langage machine est lui aussi très simple . Ceux qui ne possèdent pas d'ASSEMBLEUR pourront se servir du petit programme BASIC joint aux listings .

Dès que le message **"court-circuitez les broches"** s'affiche , vous devez relier les bornes 20 et 39 (ou 40) du connecteur d'extension ... Un simple toucher devrait être suffisant ! Vous entendrez (à condition de ne pas avoir coupé le BUZZER) distinctement le BEEP avec un compteur défilant !

Vous pouvez remarquer que la méthode de changement d'adresse des interruptions ne diffère pas du changement des "Hoocks" lors d'une modification du traitement des erreurs .

5.1.5.2 Un deuxième programme ...

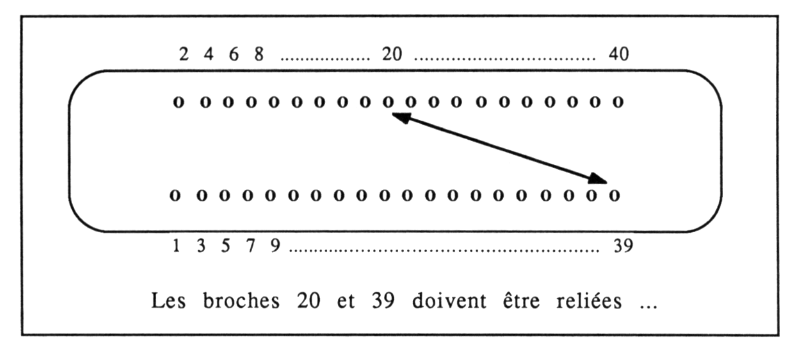
Ce deuxième exemple va traiter d'un problème très particulier : **l'interaction de deux routines écrites en langage machine** .

Nous allons interrompre le traitement d'une routine LM afin d'en effectuer une autre et , dès cette deuxième routine terminée , nous reprendrons le cours de la première jusqu'à la prochaine interruption ! Tout ceci va être réalisé grâce à notre fidèle petit fil électrique .

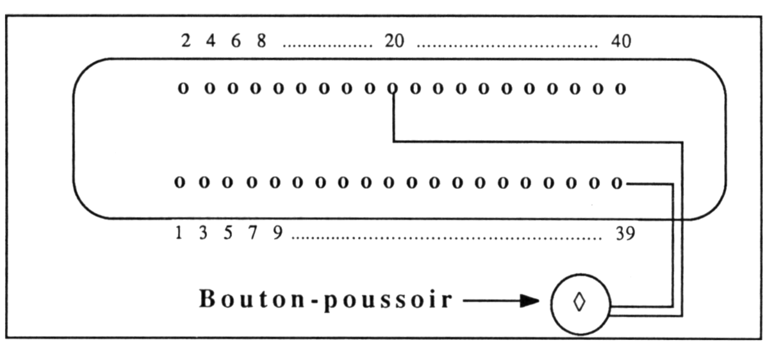
Le programme va afficher à l'écran tous les codes ASCII compris entre 0 et 255 . Lorsque vous court-circuiterez les broches 39 et 20 , un message s'affichera mais le défilement ne s'arrêtera pas pour autant !

Pour des raisons de simplification , les codes affichés s'échelonnent de 0 à 255 ... Par conséquent , ne vous étonnez pas si aucun caractère n'apparait durant les premières secondes ! De plus , une temporisation située entre les lignes 215 et 320 du listing source permet de ralentir l'affichage des caractères... Si elle n'était pas présente , vous ne pourriez rien distinguer à l'afficheur tellement la routine est véloce ! Cette routine est d'ailleurs utilisée dans le deuxième programme pour que le message puisse rester visible à l'écran suffisamment longtemps .

Nous conseillons aux amateurs de se procurer un connecteur correspondant à la prise d'extension du X-07 . **Ensuite , vous pourrez câbler un petit bouton poussoir entre les bornes 39 et 20** ... Le court-circuit à effectuer en sera d'autant facilité car vous pourrez remarquer qu'un contact trop prolongé entre les deux bornes peut fausser le bon déroulement des opérations.



BROCHAGE DU CONNECTEUR ARRIERE



SCHEMA DU BOUTON-POUSSOIR

FIGURE 5 : LES DIFFERENTS BROCHAGES

Pour ceux qui ne possèdent pas d'ASSEMBLEUR , le logiciel BASIC charge les deux parties du programme : la première détourne les interruptions et la deuxième est constituée par les deux petites routines .

Dès que le premier programme est chargé en mémoire , vous pouvez commencer à envoyer des interruptions au X-07 .

Les bricoleurs chevronnés pourront envisager par exemple de construire une petite horloge qui générerait un "top" toutes les "X-ièmes" de secondes . Ils auront la possibilité de simuler **un fonctionnement en multitâche du X-07** !

Pour cela , il suffit de placer à l'adresse de l'interruption le deuxième programme que l'on désire exécuter . Une solution intéressante consiste à récupérer le signal CLOCK présent sur la borne 29 et de le diviser à l'aide de bascules successives ou de compteurs afin d'obtenir un signal de fréquence moindre (de l'ordre de 10 à 5) . Ce signal sera alors à renvoyer sur la broche 20 du connecteur .

Pour ceux qui désirent améliorer le système du bouton-poussoir , le problème des effets de retour peut être résolu en introduisant une bascule dans le système .

5.1.6 CONCLUSION .

La détection d'interventions extérieures ouvre **un champ d'applications gigantesque au programmeur** : détection de niveaux , surveillance d'un appartement , détection d'une sonnerie téléphonique en vue de l'édification d'un serveur , etc ...

L'emploi des interruptions reste un moyen relativement simple pouvant être dédié à la résolution des problèmes liés à l'interfaçage des périphériques rapides . On peut noter que l'interruption NMI est très utilisée dans le but de relier le microprocesseur à un lecteur de disquettes ou à un circuit d'accès direct en mémoire (DMA) .

Les personnes ne se sentant pas une âme bricoleuse peuvent quand même simuler le fonctionnement multitâche en utilisant l'interface vidéo . Mais attention ! La durée du programme d'interruption est critique et si la X-720 envoie une impulsion avant que le programme ne soit terminé , le RESET général est à prévoir !!

```
5 REM 2 PRG EN LM
10 '[
20 'ORG $1A00
30 '***********************************
40 '**        CHent du saut du NMI        *
50 '***********************************
60 'LD A.$C3
70 'LD ($66).A
80 'LD HL.#DB :*DEBUT DU 2eme PROG.
90 'LD ($67).HL
100 'RET
110 '***********************************
120 '**          le 1er programme          *
130 '***********************************
140 'ORG $1B00
150 'LD A.$00
160 '#B1 INC A
170 'LD HL.$0502
180 'LD ($B8).HL
190 'CALL $C1BE
200 'CALL #TP
210 'JR #B1
215 '  ** TEMPO **
220 '#TP PUSH AF
230 'PUSH BC
240 'LD A.$7F
250 '#B2 LD B.$FF
260 '#B3 DEC B
270 'JR NZ.#B3
280 'DEC A
290 'JR NZ.#B2
300 'POP BC
310 'POP AF
320 'RET
330 '***********************************
340 '**         le 2eme programme          *
350 '***********************************
360 '#DB PUSH AF
370 'PUSH BC
380 'PUSH DE
390 'PUSH HL
400 'LD HL.$0101
410 'LD ($B8).HL
420 'LD HL.#ME
430 'CALL $FEF7
440 'CALL #TP
450 'CALL $CE9E
460 'POP HL
470 'POP DE
480 'POP BC
490 'POP AF
500 'RETI
510 '#ME DEFM INTERUPTION !!!!
520 'DEFB $00
530 ']
```

FIGURE 6 : LES 2 PROGRAMMES

```
10 REM LES 2
20 FOR I=&H1A00 TO &H1A0B
30 READ A$:POKE I,VAL("&H"+A$):NEXT
50 RESTORE  700
60 FORI=&H1B00 TO &H1B4C
70 READ A$:POKEI,VAL ("&H"+A$):NEXT
90 EXEC&H1A00:EXEC&H1B00
500 DATA3E,C3,32,66,00,21,20,1B,22
501 DATA67,0,C9
700 DATA3E,0,3C,21,2,5,22,B8,0
701 DATACD,BE,C1,CD,11,1B
710 DATA18,F1,F5,C5,3E,7F,6,FF,5
711 DATA20,FD,3D,20,F8
720 DATAC1,F1,C9,F5,C5,D5,E5,21,1
721 DATA1,22,B8,0,21
730 DATA3C,1B,CD,F7,FE,CD,11,1B,CD
731 DATA9E,CE,E1,D1
740 DATAC1,F1,ED,4D,49,4E,54,45,52
741 DATA55,50,54,49,4F,4E,20,21
750 DATA21,21,21,0
```

FIGURE 6 : FIN

```
10 '[
20 'ORG $1B00
30 '*********************************
40 '**      CHent du saut pour NMI      *
50 '*********************************
60 'LD A.$C3
70 'LD ($66).A
80 'LD HL.#DB
90 'LD ($67).HL
100 'RET
110 '*********************************
120 '**       LA ROUTINE DE BEEP        *
130 '*********************************
140 '#DB LD A.$07:*"BELL=CHR$(07)
150 'CALL $C1BE
160 'RETI
170 ']
```

FIGURE 7 : LE BEEP !!

```
5 REM LE BEEP
10 FOR I=&H1B00 TO &H1B12
20 READ A$:POKE I,VAL("&H"+A$)
30 NEXT
40 CLS:PRINT"COURT-CIRCUITEZ LES
  BROCHES"
50 FOR I=1TO10000
60 LOCATE10,2:PRINTI
70 NEXT
100 DATA3E,C3,32,66,00,21,C,1B
101 DATA22,67,0,C9
102 DATA3E,7,CD,BE,C1,ED,4D
```

FIGURE 7 : FIN

5.2 LES ENTREES/SORTIES DU X-07 .

Une des particularités du CANON est représentée par **son exceptionnelle ouverture vers le monde extérieur** . En effet , quatre prises très importantes trônent sur trois des côtés de notre machine : la prise magnétophone , série , parallèle et la prise d'extension .

On peut facilement utiliser le port d'extension du bus du X-07 car l'Assembleur Z-80 comprend parfaitement **les ordres IN et OUT** permettant d'utiliser cette prise .

Par contre , les prises série et magnétophone sont beaucoup plus mals loties ... Leur programmation passe en effet par **l'ACIA** et nécessite de ce fait une très bonne connaissance de l'organisation interne du CANON .

Il existe pourtant une solution intéressante pour simplifier l'utilisation de ces différents ports : il suffit d'employer les routines ROM correspondant à la fonction INIT #n afin d'ouvrir un fichier d'entrées/sorties . Dans ces conditions, plus rien ne s'oppose à une utilisation très performante du X-07 en matière d'entrées/sorties !!

En effet , l'utilisation simultanée de la prise série et du Langage Machine va permettre de réaliser quantité d'applications originales ... Tenez-vous prêt !!

5.2.1 LES DIVERS DISPOSITIFS .

Avant de passer en revue les divers dispositifs d'entrées/sorties existant sur le X-07 , notons que si l'entrée de mots se fait toujours à une vitesse fixée par le dispositif relié au X-07 , la durée entre l'entrée de deux mots peut être considérablement réduite ...

5.2.1.1 Les dispositifs assignables par l'ordre INIT #n .

Il existe cinq dispositifs directement assignables par l'ordre INIT #n :

* GPR : ce dispositif correspond à l'imprimante graphique . Lorsqu'il est initialisé , la taille des caractères est fixée à 2 et les codes "retour chariot/fin de ligne" sont envoyés à la X-710 .
* LPT : ici , seuls les codes "retour chariot/fin de ligne" sont envoyés sur la prise parallèle (Centronics) .
* KBD : la mémoire tampon du clavier est remise à 0 .
* MEV : le fichier dont le nom est spécifié est recherché . S'il existe , les pointeurs de lecture et d'écriture sont placés au début du fichier . La taille étant alors ignorée , il est donc inutile de la spécifier . Par contre , si le fichier n'existe pas , il est créé avec la taille spécifiée .
* CON : ce dispositif ne fait strictement rien .

5.2.1.2 Les autres dispositifs .

Ces dispositifs sont au nombre de cinq : CASI , CASO , OPT , COM , PRT .

Dans ces différents cas , on assigne à une des sorties spécifiées le dispositif d'entrées/sorties désiré . On règle le générateur de Bauds à la vitesse désirée ainsi que le mode de transmission .

Notons que toutes ces fonctions ne sont réalisées que pour le dispositif COM . En effet , les réglages de cadence de transmission et de mode se font automatiquement pour les quatre autres systèmes .

5.2.1.3 L'initialisation .

Tous les dispositifs sont initialisés de la même façon à l'aide de l'appel à un sous-programme unique . Il faut impérativement spécifier lors d'un appel le dispositif que l'on désire utiliser .

Lors de l'appel à ce sous-programme , le X-07 va rechercher dans une table à quel dispositif il doit accéder . Dès qu'il a résolu cette première phase , il programme ses interfaces de la manière demandée . Notons qu'une recherche dans la mémoire morte permet de retrouver les noms des dispositifs initialisables .

Cette table constitue un catalogue des systèmes existants . Prenons l'exemple de "CASI" ; à l'adresse E782h , nous allons trouver en hexadécimal les données suivantes : 43 41 53 49 BA 00 02 12 E5 0A E0 35 DE 0F E1

* 43 : C	* BA : :	* 12 , E5 , 0A , E0 , 35 , DE , 0F , E1 :
* 41 : A	* 00 : N° du	données correspondant aux sous-
* 53 : S	* 02 : dispositif	programmes liés au dispositif .
* 49 : I		

C'est en précisant l'adresse du numéro du dispositif que nous pouvons spécifier le dispositif que l'on désire mettre en oeuvre lors de l'appel au sous-programme d'initialisation commun à tous les systèmes .

Pour cela , nous devons utiliser une partie spécialisée de la zone système du X-07 située à l'adresse **2C5h** . Lors de l'ouverture du dispositif , on emploiera le registre DE afin de pointer sur le numéro du dispositf à initialiser. Le X-07 placera alors à l'adresse 2C5h l'adresse du dispositif dans le tableau et sauvegardera les registres HL , BC et DE à chaque fois que l'on fera appel au système en question .

Notons que cinq dispositifs peuvent être ouverts au maximum . En effet , la zone allant de **2C5h à 2E8h** ne peut contenir que 40 octets : deux octets pour l'adresse du tableau et six octets pour les trois registres à sauver et ceci , cinq fois seulement . **Voilà pourquoi on ne peut utiliser un numéro de fichier supérieur à 5** .

La table en ROM est codée de cette manière :

_ Nom du dispositif .
_ Numéro du dispositif .
_ Deux octets pour le sous-programme de sortie d'un octet .
_ Deux octets pour le sous-programme d'entrée d'un octet .
_ Deux octets pour le sous-programme d'initialisation .
_ Quelques octets sont laissés pour un sous-programme optionnel .

Après avoir passé en revue tous les dispositifs existants , nous allons pouvoir dans l'étude de routines d'entrées/sorties ...

5.2.2 LES FICHIERS CASO & CASI .

Nous allons réaliser une petite routine de sauvegarde et de récupération de données sur K7 . Pour réaliser cette application , nous avons besoin de plusieurs choses :

_ Initialiser le fichier de sortie pour la sauvegarde .
_ Déterminer un "délimiteur" qui nous indiquera le début et la fin de chaque donnée .
_ Initialiser le fichier d'entrée pour recharger les données .
_ Un test vérifiant que nous sommes bien en présence du début des données .
_ Une mémoire tampon nécessaire à la sauvegarde des données lues .
_ Un test vérifiant que nous sommes bien en présence de la fin des données .

De plus , nous devrons dans ce cas signaler une erreur d'entrée si un problème survenait . Nous utiliserons dans ce but les indicateurs du registre F (C et F) nous indiquant la présence d'une erreur .

Enfin , nous devons tenir compte du fait que la routine ROM d'entrée de données n'attend pas les données : une boucle de temporisation est donc fortement conseillée .

5.2.2.1 Le programme de sortie des données .

L'adresse de tableau de CASO est **E796h** . Nous devons donc placer cette valeur dans le registre DE lors de l'appel au sous-programme d'ouverture de fichier (Adresse E6A8h) . Nous obtenons donc le listing de la page suivante ...

```
10 ' [
20 ' ORG $1C00
30 ' LD DE.$E987              ; Adresse de tableau du dispositif CASO .
40 ' LD IY.$02C5              ; Adresse de la partie spécialisée de la RAM .
50 ' LD A.$00                 ; Mise à 0 du registre A .
60 ' CALL $E6A8               ; Appel d'ouverture du dispositif .
70 ' RET
80 ' ]
```

Si vous faites un PRINT #1,"A" , le X-07 émettra le son caractéristique d'une sauvegarde de K7 , ce qui confirme bien que la sortie K7 a bien été initialisée .

Il nous faut ensuite démarrer le magnétophone car celui ci n'est pas automatiquement commandé par la routine de sortie de caractères à laquelle nous allons faire appel . Pour ce faire , vous devez inscire le chiffre 1 dans la bit de poids faible du port de sortie F4h , en faisant OUT ($F4),1 .

Il est maintenant nécessaire de laisser au magnétophone le temps de stabiliser sa vitesse . Une petite routine de temporisation conviendra parfaitement à cet usage ... Elle utilise la paire de registres BC . Elle sera appelée autant de fois que nécessaire .

De plus , nous ferons précéder la sortie de nos données par trois chiffres "0" afin de pouvoir les retrouver à la fin et nous les ferons suivre par trois "$FF" . Lors de la sortie des données , le double registre HL est employé afin de pointer sur les données à sortir . Voici le résultat de nos périgrinations :

```
10 ' [
20 ' ORG $1C00
30 ' LD DE.$E796
40 ' LD IY.$02C5
50 ' LD A.$00
60 ' CALL $E6A8
65 ' IN A.($F4)               ; Le démarrage du magnétophone nécessite la
66 ' SET 0.A                  ; mise à 1 du bit D0 du port F4 (lignes 65 à
70 ' OUT ($F4).A              ; 70) .
80 ' CALL #TP                 ; Temporisation ...
90 ' CALL #TP
100 ' LD DE.$02C5
110 ' CALL $E827              ; On assigne la sortie à CASO
120 ' LD A.$00                ; Les lignes 120 à 170 signalent le début des
130 ' CALL $E88F              ; données par un triple 0
140 ' LD A.$00
150 ' CALL $E88F
160 ' LD A.$00
170 ' CALL $E88F
```

```
180 ' LD HL.#DD              ; Les lignes 180 à 230 constituent la sortie
190 ' #ST LD A.(HL)          ; des données ...
200 ' CP $00
210 ' JR Z.#TE
220 ' CALL $E88F
225 ' INC HL
230 ' JR #ST
240 ' #TE LD A.$FF           ; Les lignes 240 à 290 signalent la fin des
250 ' CALL $E88F             ; données par un triple 255 ...
260 ' LD A.$FF
270 ' CALL $E88F
280 ' LD A.$FF
290 ' CALL $E88F
291 ' CALL #TP               ; Les lignes 291 à 294 constituent l'arrêt du
292 ' IN A.($F4)             ; magnétophone ...
293 ' RES 0.A
294 ' OUT ($F4).A
300 ' RET
310 ' #TP PUSH BC            ; Les lignes 310 à 390 constituent la boucle
320 ' LD B.$00               ; de temporisation .
330 ' #E1 LD C.$FF
340 ' #E2 DEC C
350 ' JR NZ.#E2
360 ' DEC B
370 ' JR NZ.#E1
380 ' POP BC
390 ' RET
400 ' #DD DEFB $01 , 02 , 03 , 04 , 05 , 06 , 00        ; données .
410 ' ]
```

Voici le listing BASIC de cette routine :

```
10 FOR I = &H1C00 TO &H1C00 + 103
20 READ A$ : POKE I , VAL ("&H" + A$)
30 NEXT I
40 PRINT "EXEC &H1C00 POUR DEMARRER LE PROGRAMME"
50 DATA 11 , 96 , E7 , FD , 21 , C5 , 2 , 3E , 0 , CD , A8 , E6 , DB , F4 , CB ,
   C7 , D3 , F4 , CD , 54 , 1C , CD , 54 , 1C , 11 , C5 , 2 , CD , 27 , E8 , 3E , 0 ,
   CD, 8F , E8 , 3E , 0 , CD , 8F , E8 , 3E , 0 , CD , 8F , E8 , 21 , 61 , 1C , 7E ,
   FE , 0 , 28 , 6 , CD , 8F , E8 , 23 , 18 , F5 , 3E , FF , CD , 8F , E8 , 3E , FF ,
   CD , 8F , E8 , 3E , FF , CD , 8F , E8 , CD , 54 , 1C , DB , F4 , CB , 87 , D3 ,
   F4 , C9 , C5 , 6 , 7F , E , FF , D , 20 , FD , 5 , 20 , F8 , C1 , C9 , 1 , 2 , 3 , 4 ,
   5 , 6 , 0
```

De plus , on utilise le sous-programme de sortie de caractères situé en E88Fh . Celui-ci est commun à la sortie des données pour tout périphérique .

Si l'on désire travailler avec le magnétophone , le premier dispositif étant la K7 , il est impératif de faire appel au sous-programme E827h en plaçant dans le registre DE la valeur 2C5h .

Par contre , pour assigner le deuxième dispositif , on placera dans le registre DE l'adresse 2C5h + 8 (8 octets = 2 octets pour l'adresse de tableau + 6 octets pour les trois registres) , c'est à dire **2CDh** .

On peut donc initialiser plusieurs dispositifs et passer de l'un à l'autre en assignant chaque fois la sortie au dispositif voulu . On peut ainsi entrer des données sur la prise série du CANON et les sortir sur l'imprimante parallèle ou entrer des données au clavier et les sortir sur l'écran du MINITEL .

5.2.2.2 Le sous-programme de récupération des données .

Nous allons maintenant utiliser CASI pour récupérer les données sauvegardées précédemment . Pour ce faire , il faut d'abord initialiser le dispositif CASI comme dispositif d'entrée .

Ensuite , nous lirons les données parvenant sur la prise K7 jusqu'à trouver le triple 0 . Les données arrivant ensuite sur la prise seront celles recherchées et ce , jusqu'au triple FFh . Afin de simplifier le programme , nous ne tiendrons pas compte des erreurs d'entrées/sorties . Notons que les personnes désirant quand même traiter ces erreurs le peuvent ... En effet , les bits C et Z positionnés à 1 indiquent des problèmes de chargement : à vous de concocter la routine de traitement d'erreurs .

Sur le même principe que précédemment , l'adresse de tableau de CASI est **E787h** . On fait donc appel au même programme d'initialisation ...

En ce qui concerne l'entrée des données , nous ferons appel à une routine spécialisée de la ROM du X-07 située en E8D4h . Cette routine contient elle-même un appel à **la routine "ABORT"** permettant de stopper le logiciel sans passer par la touche "ON/BREAK" .

La temporisation n'est pas utile ici car les données arrivent de la K7 . Par contre , vous devez prêter attention à la durée du traitement entre l'entrée de chaque donnée ... Si ce dernier est trop long , vous risquez de rater une ou plusieurs données !

Voici le "source" de ce programme :

```
10 ' [
20 ' ORG $1C00
30 ' * INIT #1 , "CASI:"     ; Les lignes 30 à 70 initialisent le dispositif 1
40 ' LD DE.$E787             ; en tant que "CASI" .
50 ' LD IY.$2C5
```

```
60 ' LD A.$0
70 ' CALL $E6A8
80 ' * Démarrage du magnétophone
90 ' IN A.($F4)
100 ' SET 0.A
110 ' OUT ($F4).A
120 ' * On assigne CASI:
130 ' LD DE.$2C5
140 ' CALL $E827
150 ' * Entrées des données ; Le triple 0 est recherché des lignes 150 à
160 ' #E1 CALL $E8D4        ; 210 ...
170 ' JR NZ.#E1
180 ' CALL $E8D4
190 ' JR NZ.#E1
200 ' CALL $E8D4
210 ' JR NZ.#E1
220 ' LD HL.$1B00           ; Les lignes 220 à 280 constituent la
230 ' #E2 CALL $E8D4        ; recherche de la fin des données ....
240 ' CP $FF
250 ' JR Z.#FI
260 ' LD (HL).A
270 ' INC HL
280 ' JR #E2
290 ' #FI CALL $E8D4        ; Si le code suivant le premier $FF n'est pas
300 ' CP $FF                ; $FF , on écrit le code dans la mémoire
310 ' JR Z.#F1              ; (lignes 290 à 320) .
320 ' LD A.(HL)
330 ' INC HL                ; On incrémente le pointeur et on continue le
340 ' JR #E2                ; chargement ...
350 ' #F1 CALL $E8D4        ; Si le deuxième code est $FF , on vérifie le
360 ' CP $FF                ; troisième ... lignes 350 à 400 .
370 ' JR Z.#F2
380 ' LD A.(HL)
390 ' INC HL
400 ' JR Z.#E2
410 ' * Arrêt magnétophone
420 ' #F2 IN A.($F4)
430 ' RES 0.A
440 ' OUT ($F4).A
450 ' RET
```

Voici le listing BASIC de cette routine :

```
10 FOR I = &H1C00 TO &H1C00 + 82
20 READ A$ : POKE I , VAL ("&H" + A$)
30 NEXT
40 PRINT "EXEC &H1C00 pour démarrer ..."
50 DATA 11 , 87 , E7 , FD , 21 , C5 , 2 , 3E , 0 , CD , A8 , E6 , DB , F4 , CB ,
   C7 , D3 , F4 , 11 , C5 , 2 , CD , 27 , E8 , CD , D4 , E8 , 20 , FB , CD , D4 , E8 ,
   20 , F6 , CD , D4 , E8 , 20 , F1 , 21 , 0 , 1B , CD , D4 , E8 , FE , FF , 28 , 4 ,
   77 , 23 , 18 , F5 , CD , D4 , E8 , FE , FF , 28 , 4 , 7E , 23 , 18 , EA , CD , D4 ,
   E8 , 8F , FE , FF , 28 , 4 , 7E , 23 , 18 , DF , DB , F4 , CB , 87 , D3 , F4 , C9
```

Les méthodes employées pour tester le début et la fin de l'enregistrement peuvent vous paraître un peu rustiques mais la rapidité prime . On préfèrera donc faire les tests les uns après les autres plutôt que d'utiliser une boucle . Le calcul de saut pour le microprocesseur risquerait de vous faire perdre des informations .

Les données entrées se retrouvent après l'exécution de cette routine les unes à la suite des autres à partir de l'adresse **1B00h** . N'attendez pas une fiabilité exemplaire de ces deux petits programmes ... En effet , aucun test ne permet de détecter des erreurs d'entrées/sorties . Le X-07 peut très bien entrer une donnée pour une autre sans que rien ne le prédise .

Nous ne développerons pas ici de routines plus performantes ... En effet , celles-ci existent déjà dans la ROM du CANON . Mais rien ne vous empêche désormais d'employer la prise magnétophone pour vous fabriquer , par exemple , un digitaliseur très simple de sons .

5.2.3 UTILISATION DES FICHIERS GPR & KBD .

Nous allons dans ce paragraphe initialiser l'imprimante graphique comme premier dispositif . Mais afin d'éviter de tomber dans la médiocrité des répétitions intempestives , nous allons initialiser en même temps le clavier comme fichier numéro 2 . De cette façon , le X-07 recopiera sur l'imprimante toutes les touches pressées sur son clavier .

Nous avonc choisi le clavier comme dispositif d'entrée et non la prise série car cette dernière fait l'objet d'un programme complet . Mais rien ne vous empêche de le faire en tenant compte de la transcription des signaux provenant par exemple d'un MODEM ou d'un MINITEL .

Pour réaliser cette application , il vous faut l'adresse de tableau des dispositifs (GPR --> E7B2h et KBD --> E778h) .

Nous placerons lors de l'initialisation dans le registre DE l'adresse de base de la RAM (2C5h) pour le premier dispositif et l'adresse 2CDh pour le deuxième système . De plus , l'initialisation de KBD va vider la mémoire tampon du clavier , ceci nous évitant de l'accomplir ...

Les routines d'initialisation des dispositifs ne se distinguent donc pas des précédentes mis à part pour KBD . Notons que l'on peut aussi utiliser le dispositif LPT comme système de sortie (imprimante CENTRONICS) . Dans ce but , il suffit de remplacer la valeur **E7B2h par E7F8h** . Voici le listing :

```
10 ' [
20 ' ORG $1C00                ; Les lignes 20 à 70 constituent l'initialisation
30 ' * INIT #1,"ĞPR:"         ; du dispositif 1 ...
40 ' LD DE.$E7B2
50 ' LD IY.$02C5
60 ' LD A.$0
70 ' CALL $F6A8
80 ' * INIT #2,"KBD:"         ; Les lignes 80 à 120 représentent l'initialisa-
90 ' LD DE.$E778              ; tion du dispositif n°2 .
100 ' LD IY.$02CD
110 ' LD A.$0
120 ' CALL $E6A8
130 ' * Assignation de KBD    ; Entrées sur la console ...
140 ' #BL LD DE.$2CD
150 ' CALL $E827
170 ' CALL $E8D4              ; Entrée d'une touche ...
175 ' PUSH AF
190 ' LD DE.$2C5              ; Les sorties se font sur l'imprimante ...
200 ' CALL $E827
205 ' POP AF
206 ' CP $0E                  ; Si on appuie sur CTRL N , on arrête .
207 ' RET Z
210 ' CALL $E88F              ; Sinon , écriture sur l'imprimante .
220 ' JR #BL
230 ' ]
```

Voici le listing BASIC de cette routine :

```
10 FOR I = &H1C00 TO &H1C00 + 48
20 READ A$ : POKE I , VAL ("&H" + A$) : NEXT
40 PRINT "EXEC &H1C00 pour démarrer ..."
50 DATA 11 , F8 , E7 , FD , 21 , C5 , 2 , 3E , 0 , CD , A8 , E6 , 11 , 78 , E7 ,
   FD , 21 , CD , 2 , 3E , 0 , CD , A8 , E6 , 11 , CD , 2 , CD , 27 , E8 , CD , D4 ,
   E8 , F5 , 11 , C5 , 2 , CD , 27 , E8 , F1 , FE , E , C8 , CD , 8F , E8 , 18 , E7
```

Ce programme pourrait être plus rapide en utilisant directement le port F1h sur lequel se retrouvent les touches pressées . Mais le but de cette routine étant d'employer les routines d'entrées/sorties , ceci se révélerait inutile !

De plus , cela nous conduirait à tester si les données se trouvent bien sur le port **F1h** alors que la routine **E8D4h** gère automatiquement cela . En effet , elle ne rend la main que si une donnée est présente .

5.2.4 UTILISATION DE LA PRISE SERIE .

Avec le contenu de ce paragraphe, vos routines vont prendre le Concorde pour Rio de Janeiro !!! En effet , en langage machine , les possibilités de traitement deviennent grandioses car le temps séparant l'entrée et la sortie de deux mots peut être considérablement réduit !

Nous allons nous limiter à deux programmes , l'un servant à écrire sur le MINITEL et l'autre servant à y lire ...

Lors de l'initialisation du dispositif , il est important d'indiquer la vitesse de transmission et le mode ACIA . Par conséquent , nous allons employer deux registres du NSC 800 ... La vitesse de transmission sera chargée dans le registre IX et le mode ACIA dans le registre B . De plus , l'adresse de tableau du dispositif COM est **E7A4h** .

Le premier programme inscrira à l'écran du MINITEL un message après avoir effacé l'afficheur . Dans cette optique , on pointe le registre HL sur le message . Après l'affichage de chaque lettre , si le caractère suivant est un 0 , le message sera terminé et la routine s'arrêtera .

Pour effacer l'écran , il suffit d'envoyer **le code 12** qui correspond à la commande CLS du MINITEL . On utilise toujours le même dispositif d'initialisation commun à tous les systèmes .

Voici le listing de cette routine :

```
10 ' [
20 ' ORG $1C00
30 ' LD DE.$E7A4          ; Dispositif COM activé
40 ' LD A.$0
50 ' LD B."G"             ; Mode G de l'ACIA
60 ' LD IX.&1200          ; Vitesse : 1200 bauds
70 ' LD IY.$2C5
80 ' CALL $E6A8           ; Ouverture ...
85 ' LD DE.$2C5
86 ' CALL $E827
87 ' LD HL.#ME            ; Ecriture du message ...
90 ' #ET LD A.(HL)
```

```
91 ' CP $0
92 ' JR Z.#FI
100 ' CALL $E88F
105 ' INC HL
106 ' JR #ET
120 ' #FI RET
124 ' #ME DEFB &12
125 ' DEFM Cela marche bien comme cela ...
126 ' DEFB $00
130 ' ]
```

Voici le listing BASIC :

```
20 FOR I = &H1C00 TO &H1C00 + 63
30 READ A$ : POKE I , VAL ("&H" + A$) : NEXT
50 PRINT " EXEC &H1C00 pour démarrer ..."
60 DATA 11 , A4 , E7 , 3E , 0 , 06 , 47 , DD , 21 , B0 , 4 , FD , 21 , C5 , 2 ,
   CD , A8 , E6 , 11 , C5 , 2 , CD , 27 , E8 , 21 , 27 , 1C , 7E , FE , 0 , 28 , 6 ,
   CD , 8F , E8 , 23 , 18 , F5 , C9 , C , 43 , 41 , 20 , 4D , 41 , 52 , 43 , 48 ,
   45 , 20 , 42 , 49 , 45 , 4E , 20 , 43 , 4F , 4D , 4D , 45 , 20 , 43 , 41 , 0
```

Occupons-nous pour finir ce paragraphe de la lecture des données provenant d'un périphérique sériel . Cela occasionne peu de changements . En effet , l'initialisation demeure identique . Le changement réside dans l'appel de la routine de lecture au lieu de celle de l'écriture , la donnée se trouvant alors dans le registre A . Un appel à l'adresse C1BEh fait d'ailleurs apparaître cette donnée . Voici le listing de cette routine :

```
20 ' [
30 ' ORG $1C00
40 ' LD DE.$E7A4          ; Les lignes 40 à 90 initialisent le1er fichier
50 ' LD A.$0              ; à COM .
60 ' LD B."G"
70 ' LD IX.&1200
80 ' LD IY.$2C5
90 ' CALL $E6A8
100 ' LD DE.$2C5          ; Les lignes 100 à 110 assignent l'entrée .
110 ' CALL $E827
120 ' CALL $E8D4          ; Caractère entré dans A .
130 ' CALL $C1BE          ; Caractère affiché à l'écran .
140 ' RET
150 ' ]
```

Voici le listing BASIC :

```
20 FOR I = &H1C00 TO &H1C00 + 29
30 READ A$ : POKE I , VAL ("&H" + A$) : NEXT
50 PRINT " EXEC &H1C00 pour démarrer ..."
60 DATA 11 , A4 , E7 , 3E , 0 , 6 , 47 , DD , 21 , B0 , 4 , FD , 21 , C5 , 2 , CD ,
    A8 , E6 , 11 , C5 , 2 , CD , 27 , E8 , CD , D4 , E8 , CD , DE , C1 , C9
```

5.2.5 CONCLUSION .

Pour conclure ce chapitre , nous vous présentons le "DUMP" de la table mémoire où se trouvent placées l'adresse des sous-programmes des dispositifs et la liste d'adresses de tableau de chaque système .

Dispositif	Adresse en hexa.	Commentaires
RAM	E7CE	Mémoire RAM
CON	E7EA	Console
KBD	E778	Clavier/écran LCD
GPR	E7B2	Imprimante graphique
LPT	E7F8	Imprimante CENTRONICS
CASI	E787	Entrée K7
CASO	E796	Sortie K7
OPT	E7DC	Coupleur optique
COM	E7A4	Prise entrées/sorties série
PRT	E7C0	Imprimante série

Après toutes ces émotions , nous allons pouvoir attaquer la X-720 ... Nous espérons que ce chapitre très important concernant les interruptions et les entrées/sorties vous aura apporté toutes les solutions dont vous réviez depuis si longtemps !!

Pour le moment , cap sur la vidéo "made in CANON" ...

L' INTERFACE X-720

Ce sixième chapitre est entièrement consacré à un périphérique relativement controversé : **la X-720** ou , en termes simples , **l'interface péritélévision** . Comme son nom l'indique , elle permet de relier le X-07 à un poste équipé de la prise péritel ...

Assez encombrante et non autonome , elle ajoute un regrettable "fil à la patte" au X-07 qui perd alors sa vocation de portable pour se consacrer au familial ... Malheureusement , le peu de logiciels , sur le marché , adaptés à cette interface n'ont pas révélé les intéressantes possibilités dont elle est pourtant dotée .

Nous allons à travers deux approches opposées (le HARD et le SOFT) vous exposer quelques mystères de la X-720 ...

6.1 L' APPROCHE "HARD" .

6.1.1 Généralités .

La X-720 est architecturée autour d'un générateur d'affichage vidéo de la célèbre firme MOTOROLA , **le MC 6847** . Mais ce processeur n'est pas seul : de la RAM et de la ROM se partagent ses faveurs !

La mémoire vive de six kilo-octets représente la capacité maximale du MC 6847 . Elle est présente sous la forme de trois circuits mémoire de deux kilo-octets chacun .

En ce qui concerne **la mémoire morte**, deux boitiers se partagent les fonctions . L'un contient les caractères affichables à l'écran et l'autre un interpréteur BASIC donnant accès à de nouvelles fonctions .

Une étude du diagramme de la X-720 met clairement en évidence la manière dont sont stockées les données présentes à l'affichage ... Deux des circuits de la mémoire vive contiennent l'état d'un point tandis que le dernier circuit contient sa couleur .

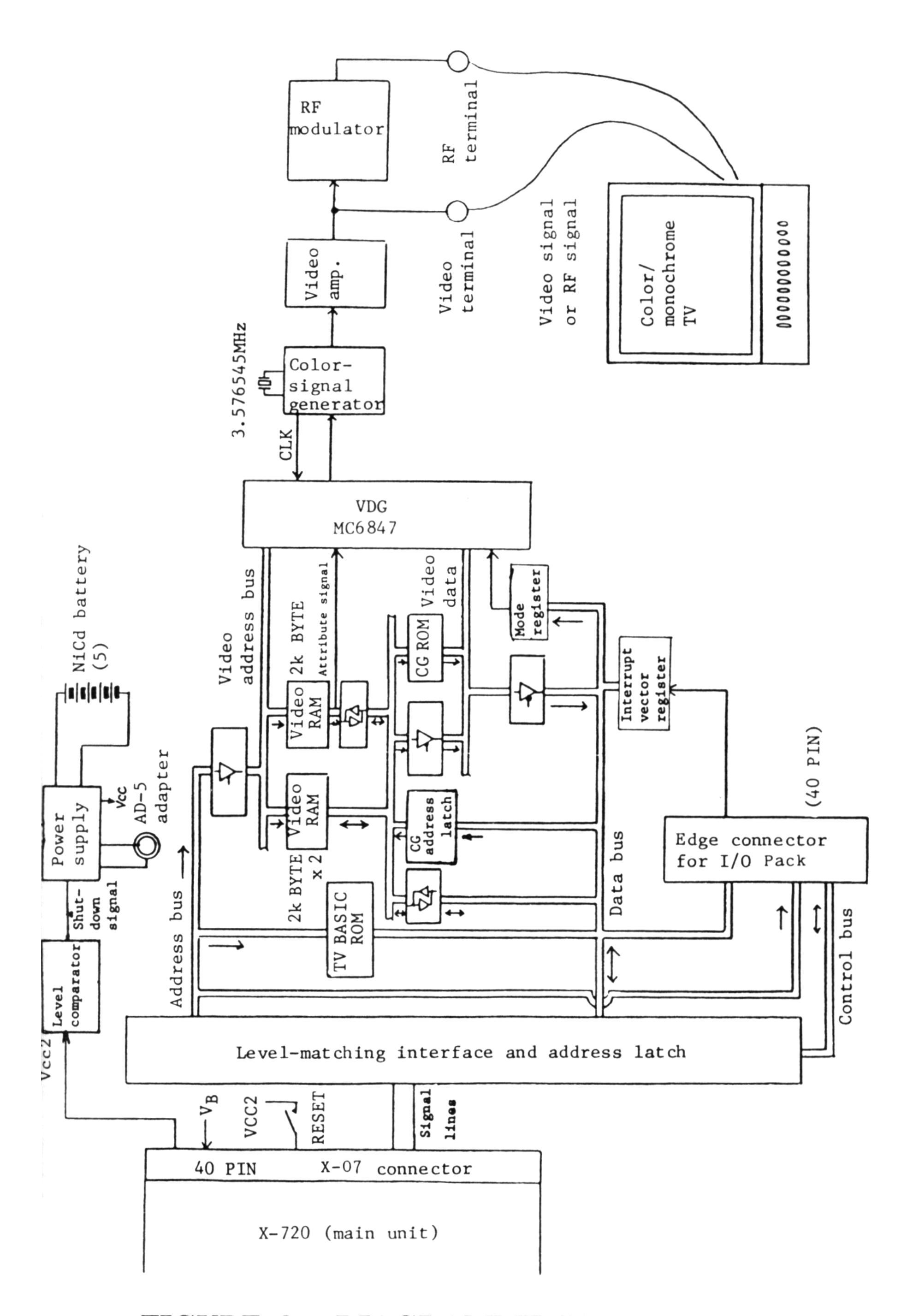


FIGURE 8 : DIAGRAMME DE LA X-720

Ad.	Uti.	LECTURE								ECRITURE							
		D7	D6	D5	D4	D3	D2	D1	D0	D7	D6	D5	D4	D3	D2	D1	D0
80-8F	C.G ROM	lecture des informations								adresse du C.G. ROM							
90-97	----	FS	INT			B4	B3	B2	B1	FSR	A/G	+1K	GM2	GM1	GM0	CSS1	CSS0
98-9F	E/S									adresse basse d'une cartouche d'entrées/sorties							

FIGURE 9 : TABLE DES E/S

6.1.2 La communication .

La communication entre le X-07 et la X-720 s'effectue **via les ports d'entrée/sortie** pour les échanges avec le MC 6847 . Par contre , les bus d'adresses et de données se chargent des communications avec la mémoire vive (VRAM) .

Le plan détaillé de l'utilisation des ports d'entrée/sortie employés pour communiquer avec le X-07 est représenté par la figure numéro 9 . Voici les détails :

Adresses 80 à 8F : le processeur utilise ces adresses pour lire **la CGROM**... Précisons que la CGROM constitue la ROM contenant le générateur de caractères. En fait , il s'agit d'une mémoire morte stockant les caractères sous forme binaire : le chiffre 1 signifie alors un point allumé et le chiffre 0 un point éteint . Nous pouvons remarquer que cette méthode est en tout point identique à la méthode de programmation des caractères redéfinissables .

Adresses 90 à 97 (en lecture) : les bits B1 à B4 ne sont pas utilisés . Par contre le bit **FS** mis à l'état bas permet au processeur d'effectuer un scrolling de l'écran . De plus , le bit **INT** constitue un bit d'interruption utilisé pour l'affichage du curseur .

Adresses 90 à 97 (en écriture) : le bit **FSR** interdit ou autorise les interruptions d'affichage du curseur (lorsque ce bit est à 0 , le curseur n'est pas affiché) . Le bit **A/G** est un drapeau de mode graphique . Le bit **+1K** modifie l'adresse de la VRAM (Ex : modification de l'adresse de la page vidéo affichée) . Les bits **GM0 à GM2** constituent les drapeaux de sélection des modes graphiques . Enfin , les bits **GSS0 et GSS1** sont des drapeaux divers .

6.1.3 REPARTITON DE LA MEMOIRE VIDEO .

La répartition de la VRAM se fait en fonction des différents modes d'écran . A cause des divers modes de résolution et de la présence de plusieurs pages d'écran , l'adressage de la mémoire vidéo diffère d'un cas sur l'autre .

En fait , pour chaque affichage , une partie d'un des trois circuits est utilisée . Les schémas présentés un peu plus loin vont vous permettre de reconnaître la mémoire occupée par chaque type de définition : les zones hachurées représentent la mémoire utilisée .

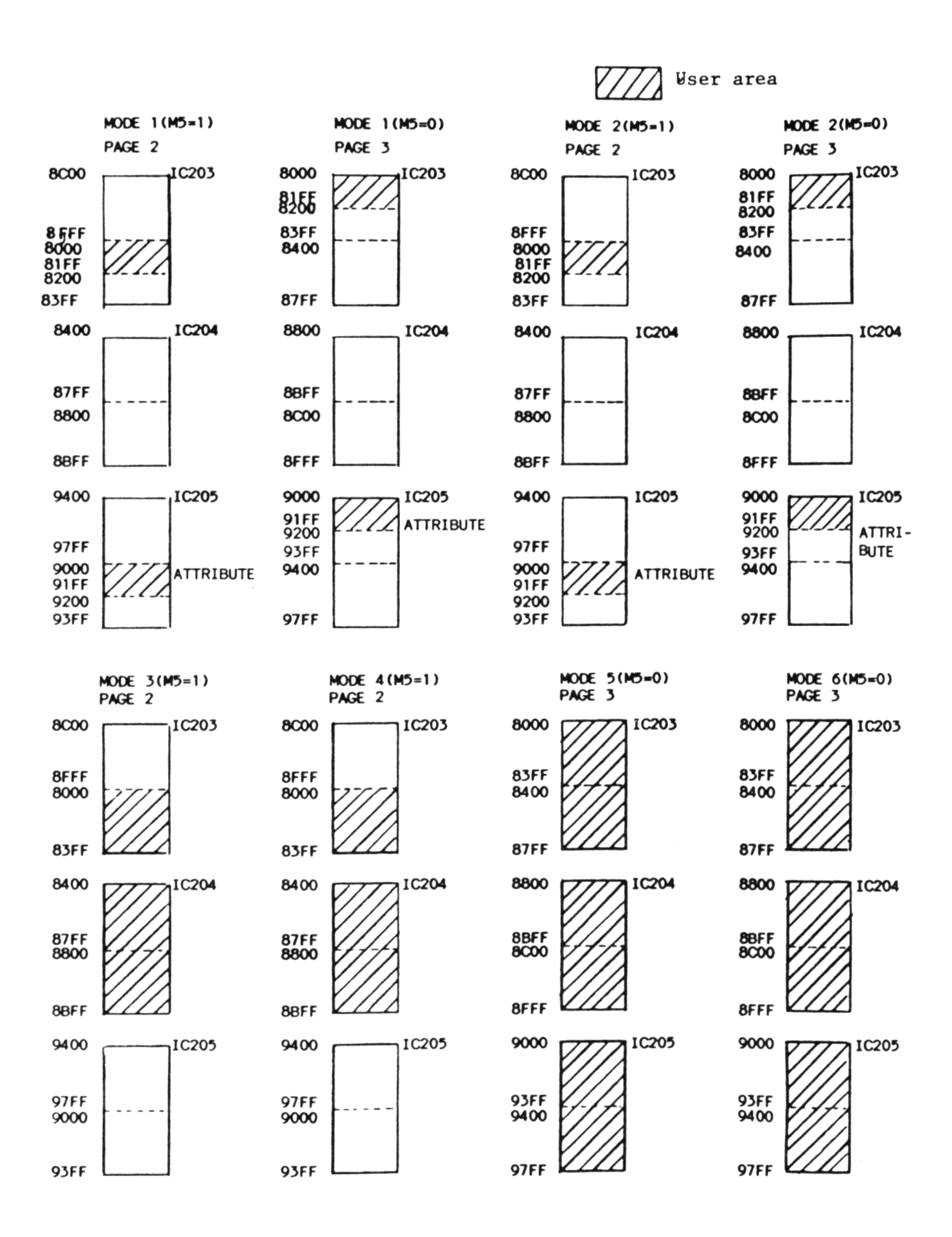


FIGURE 10 : MEMOIRE OCCUPEE

En fait , à la lecture de ces schémas , il s'avère que l'adresse rencontrée par le processeur **ne varie pas** d'une page graphique à l'autre ... En effet , le signal +1K (adresses 90 à 97 des entrées/sorties) est employé dans le but de modifier l'espace mémoire adressé . Les trois circuits de deux kilo-octets sont divisés en six segments de un kilo-octet chacun . Par conséquent , le passage d'une partie du circuit à une autre se réalise grâce au signal +1K provoquant ainsi le changement immédiat de l'image affichée à l'écran .

On devine la complexité de la gestion d'un tel système ... En effet , pour afficher un point ou une lettre à l'écran , l'interpréteur doit calculer l'adresse des valeurs à modifier en fonction du mode et de la page écran : **ceci explique la grande lenteur de l'interface X-720 !**

De plus , cette réelle complexité rend très ardue l'utilisation des nombreuses routines qui existent dans la mémoire morte du X-07 . La programmation en langage machine de la vidéo , en faisant appel à ces routines n'offre donc que peu d'intérêt , étant donné la lenteur d'exécution de celles-ci . Il est préférable de se placer dans un mode donné et de créer des fonctions SET et RESET pour ce mode particulier . Vous pouvez utilement vous référer au programme "Le piège - version vidéo" présent au dernier chapitre : il a été écrit en tenant compte de ces remarques .

Avant de passer au SOFT proprement dit , les electroniciens pourront admirer le plan détaillé du circuit de cette interface .... Bon courage !!

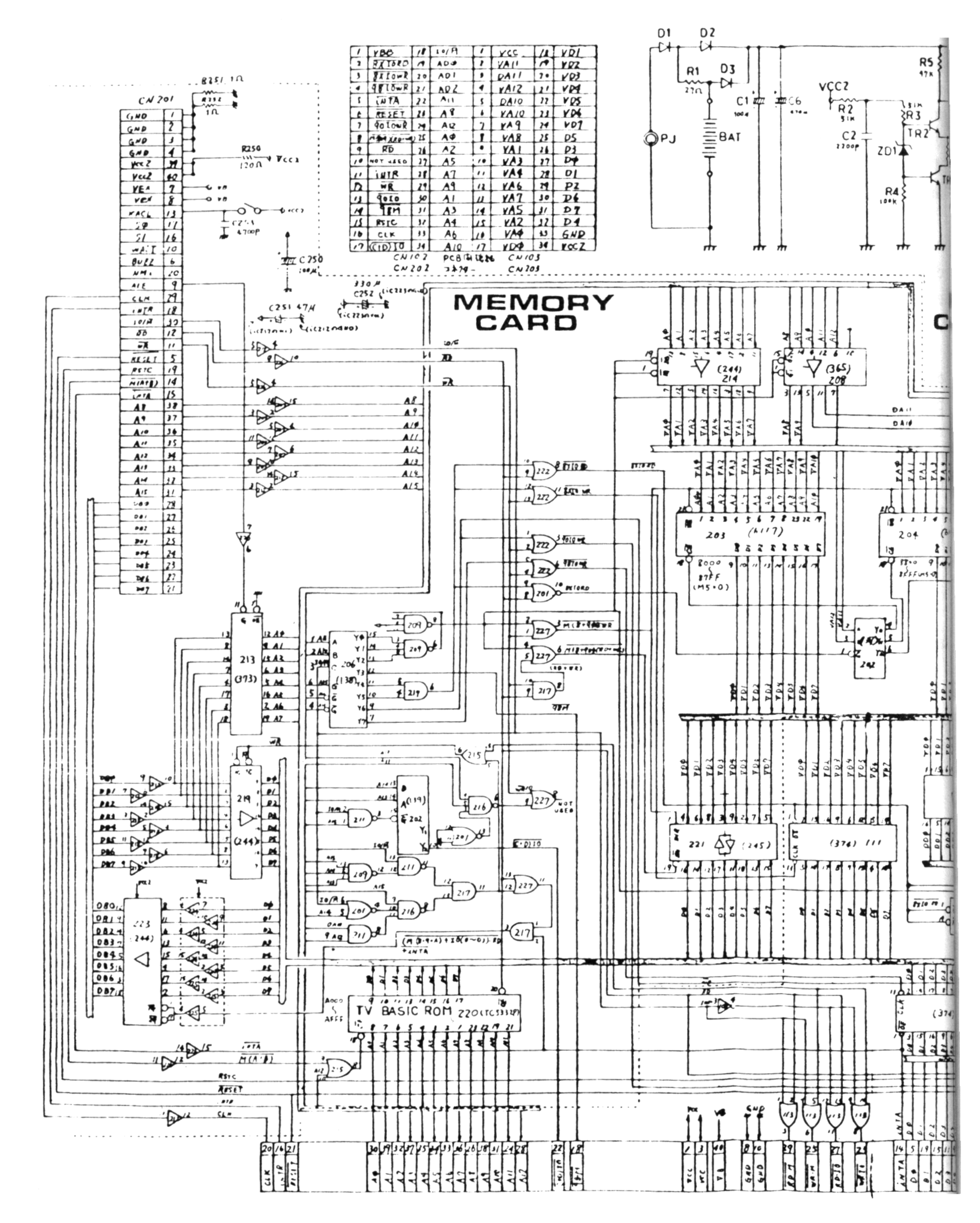


FIGURE 11 : PLAN DE LA X-720

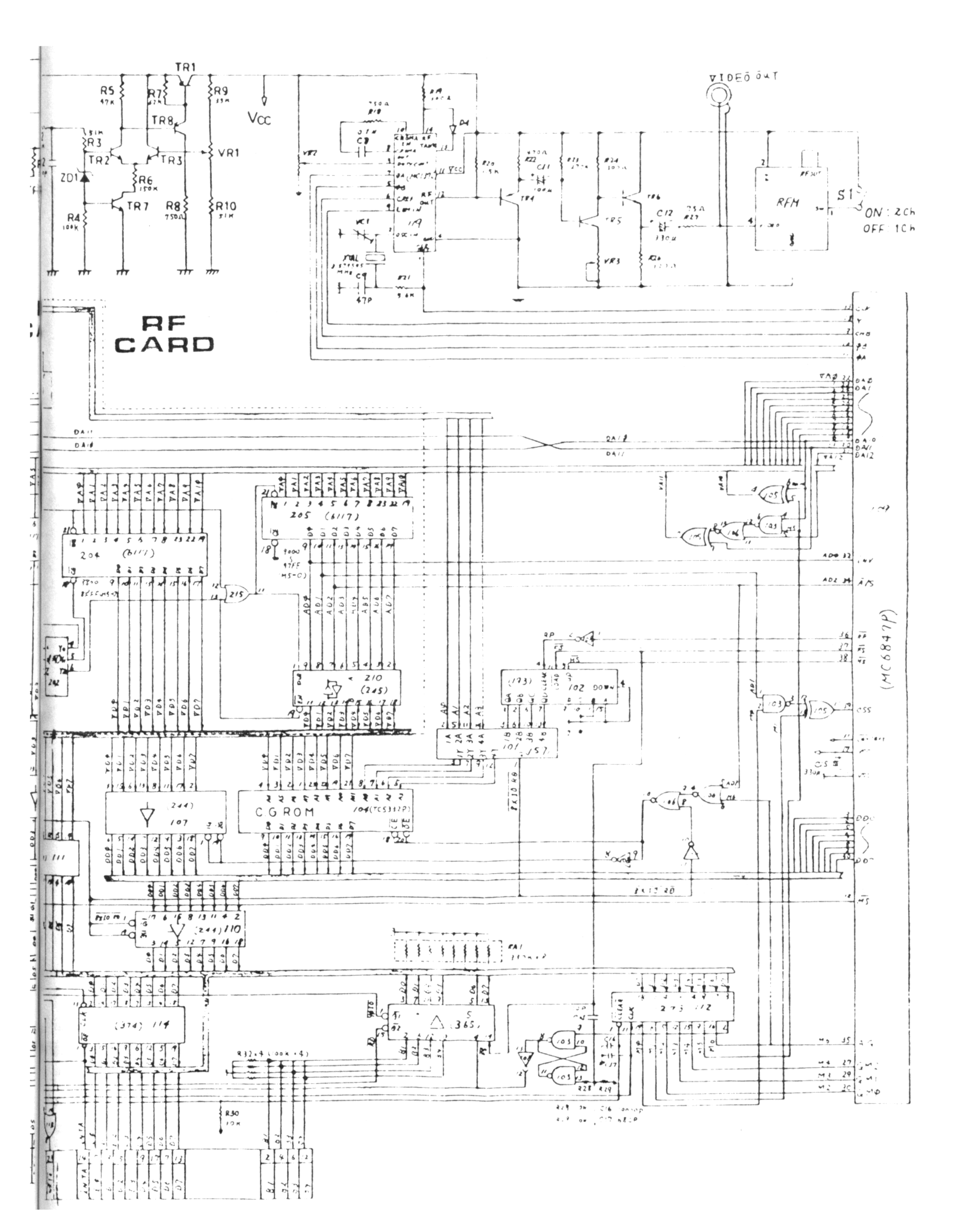


FIGURE 11 : PLAN DE LA X-720

6.2 L' APPROCHE "SOFT" .

L'interface X-720 possède donc six kilo-octets de mémoire vive (mémoire écran) implantée entre **8000h et 97FFh** et quatre kilo-octets de mémoire morte implantée entre **A000h et AFFFh** (complément à l'interpréteur du CANON) . Cette RAM débute par le mot-clé "love" , recherché à chaque mise sous tension par l'interpréteur .

Derrière ce mot-clé bien connu de tous les canonistes , **l'adresse A0CAh** du sous-programme d'initialisation de la zone de communication (nouvelles fonctions : PAINT , COLOR , SCREEN et modification de la plage d'action des commandes PSET , PRESET , POINT , LINE , CIRCLE) est présente .

Sans plus attendre , nous allons vous disséquer les points logiciels intéressants de cette interface ...

6.2.1 LES POINTS D' ENTREE ...

Voici les points d'entrée des fonctions BASIC utilisables avec la X-720 :

Fonctions	Adresses
SCREEN	AAD2h
SCREEN (X,Y)	ABBDh
COLOR	AA73h
PSET	ABFBh
PRESET	ABF6h
POINT	AC14h
LINE	AC86h
PAINT	AD9Bh
CIRCLE	AED2h

6.2.2 QUELQUES ADRESSES ...

Le sous-programme d'initialisation implante **dans la zone système** du X-07 quelques valeurs que nous allons vous exposer ...

Adresses	Commentaires
B6h	Page active . Si cet octet contient 0 , la page 1 est active . S'il contient 1 , la page 2 est active .
B7h	Page affichée . Si cet octet contient 0 , la page 1 est affichée . Par contre , s'il contient 1, c'est la page 2.

Adresses	Commentaires
D1h	Mode d'écran actif .
15h	Si cet octet est à 0 , les caractères sont normaux . Par contre , s'il est à 1 , on obtient les caractères LCD .
4D7h	Caractères KANA .
4E5h	Couleur de l'avant-plan .
4E6h	Couleur de l'arrière-plan .
4E7h	palette en cours (0 ou 2)
4C6h	Xi , Yi : coordonnées initiales de la fonction LINE .
4C8h	Xf , Yf : coordonnées finales de la fonction LINE .
4CAh	dX , dY : déplacement pour la fonction STEP .

6.2.3 FONCTIONS UTILISABLES .

Un désassemblage plus profond de la ROM de la X-720 a permis de mettre en lumière quelques fonctions utilisables dont voici les adresses :

Fonctions	Adresses	Commentaires
LSETCUR	A015h	Le fonctionnement de cette routine est le même que pour la routine C0A1h de l' écran LCD (Voir les Mystères) .
INTC	A050h	Clignotement du curseur .
ECRIRAM	A07Ch	Teste si l'écriture de la VRAM est en cours.
CONPUT	A08Bh	Sortie sur l'écran .
RST 1	A0CAh	Initialisation de la X-720 .
MODECR	A2C2h	Changement du mode d'écran . Si A=0 , le mode 1 est activé . Si A=1 , le mode 2 est activé .
RETLCD	A355h	Retour à l'écran LCD en mode 1,1,1 .
VIDEO	A374h	Affichage sur vidéo .
PUTCAR	A4D1h	Emission d'un caractère vers l'écran

De plus , la résolution étant différente pour chaque mode d'écran , les ingénieurs ont été obligés d'écrire des routines graphiques correspondant à ces définitions ... Le problème a été résolu en plaçant entre **A9FBh et AA73h** une table contenant l'adresse de chacun des sous-programmes d'exécution des fonctions , en fonction du mode choisi . Ces sous-programmes se trouvent d'ailleurs stockés de **A631h à A9FAh** .

6.2.4 CODAGE DES PIXELS EN FONCTION DES DIFFERENTS MODES .

Comme nous l'avons déjà souligné , les six modes possibles rendent le fonctionnement de l'interpréteur extrêmement complexe et ceci implique une grande lenteur d'exécution .

En effet , pour chaque fonction graphique , l'interpréteur doit décoder ou calculer les paramêtres puis , en fonction du mode écran et de la page active , il doit déterminer l'adresse du pixel à modifier .

Examinons de plus près le codage de ces pixels ...

Mode 1 : le texte seul est affiché . On a accès à 16 lignes de 32 colonnes soit 512 octets . **C'est le générateur de caractères** de la X-720 qui permet d'afficher à l'écran le contenu (valeur ASCII) de chaque case mémoire .

Mode 2 : du texte ou du graphisme peuvent être affichés . Le générateur de caractères intervient quand du texte est affiché . Par contre , quand du graphisme est reproduit à l'écran , il faut calculer **les coordonnées du pixel** (six pixels par case mémoire + deux bits pour la couleur du pixel) . Vous pouvez utilement vous reporter à la figure exposée à la page suivante .

Modes 3 et 5 : ces deux modes sont graphiques . Les caractères sont dessinés **pixel par pixel** ... Notons que les pixels sont adressés quatre par quatre et qu'un octet contient quatre pixels (donc , quatre couleurs possibles) . La figure présentée à la page suivante explicite ce codage .

Modes 4 et 6 : ces deux modes sont aussi graphiques . Les remarques sont les mêmes en ce qui concerne les caractères . **A chaque pixel correspond un bit de la mémoire écran** : les pixels sont donc adressés huit par huit . En mode 4 , trois kilo-octets de la mémoire écran sont utilisés . D'autre part , 512 octets des trois kilos restants sont utilisables en mode 2 . Enfin , en mode 6 , les six kilo-octets sont utilisés ; en effet , la définition est doublée .

Nous allons pouvoir utiliser nos connaissances toutes neuves de la X-720 dans de petites routines qui vont bouleverser votre écran !!

6.3 UTILITAIRES .

6.3.1 X-720 PRESENTE ? ...

La première routine teste **la présence de la X-720 dans le port d'extension de votre CANON .**

Le principe est très simple : la routine va écrire une valeur déterminée en haut de la VRAM . En relisant l'octet , si la valeur est différente , un traitement d'erreur sera effectué car la X-720 n'aura pas été connectée .

La valeur 80h est stockée à l'adresse 8000h . En relisant cette adresse , la X-720 sera connectée si la valeur lue est 80h ... Simple , non ?

6.3.2 EFFACEMENT PARTIEL DE L' ECRAN .

Ce petit programme permet **d'effacer les huit dernières lignes de l'écran** . La position du curseur est stockée aux adresses B8h et B9h comme pour le curseur de l'écran LCD .

le principe est de simuler une coupure de l'écran et d'effacer tout ce qui se trouve en dessous de cette coupure par la routine CE9Eh .

6.3.3 LA FONCTION SCREEN .

Pour simuler cette fonction , la routine **AB09h** de la ROM a été mise à contribution . En effet , la page active - 1 est stockée dans le registre D et la page visuelle - 1 est contenue dans le registre E .

Le mode écran a été placé dans le registre C et à l'adresse **D1h** .

6.3.4 LA FONCTION COLOR .

Deux méthodes sont possibles pour résoudre ce problème .

La première solution (la plus simple ...) consiste à charger le registre A avec le numéro de la couleur désirée puis d'appeler la routine COLOR par un **CALL A637h** .

La deuxième méthode , reprise par la routine exposée , permet **de changer la couleur du fond , des caractères et de la palette utilisée**... Qui a dit que nous recherchions la complication ? ...

Les listings sont présents dans les deux pages suivantes ... Ensuite , place aux applications !!

```
10 ' [
20 ' ORG $1C00
30 ' * TEST DE LA PRESENCE DE LA X-720
40 ' LD A , $80
50 ' LD ($8000) . A : * DEBUT  VRAM
60 ' LD A . ($8000)
70 ' CP $80
80 ' JP NZ . $F1AA : * SAUT EN ERREUR
90 ' ]
```

```
1C00 3E80        LD A , 80
1C02 320080      LD (8000) , A
1C05 3A0080      LD A , (8000)
1C08 FE80        CP 80
1C0A C2AAF1      JP NZ , F1AA
```

```
10 REM TEST DE LA X-720
20 FOR I = &H1C00 TO &H1C0D
30 READ A$ : POKE I , VAL ("&H" + A$)
40 NEXT I
50 DATA 3E , 80 , 32 , 00 , 80 , 3A , 00 ,
   80 , FE , 80
60 DATA C2 , AA , F1 , C9
```

FIGURE 12 : TEST

```
10 ' [
20 ' ORG $1C00
30 ' * EFFACEMENT PARTIEL DE L' ECRAN
40 ' CALL #SC : * EFFACEMENT DE LA
50 ' CALL $CE9E : * 8ème LIGNE à LA FIN
60 ' LD HL . $108
70 ' LD ($B8) . HL : * CURSEUR (C1 , L8)
80 ' RET
90 ' #SC LD A . $8 : * Ligne de roulement
100 ' LD ($BB) . A
110 ' LD A . &16 : * DERNIERE LIGNE
120 ' LD ($BC) . A
130 ' LD A . (&16-8+1) : * Nb de lignes
140 ' LD ($BD) . A
150 ' RET
160 ' ]
```

```
1C00 CD0D1C      CALL 1C0D
1C03 CD9ECE      CALL CE9E
1C06 210801      LD HL , 0108
1C09 22B800      LD (00B8) , HL
1C0C C9          RET
1C0D 3E08        LD A , 08
1C0F 32BB00      LD (00BB) , A
1C12 3E10        LD A , 10
1C14 32BC00      LD (00BC) , A
1C17 3A1000      LD A , (0010)
1C1A 32BD00      LD (00BD) , A
1C1D C9          RET
```

```
10 REM EFFACEMENT PARTIEL
20 FOR I = &H1C00 TO &H1C1D
30 READ A$ : POKE I , VAL ("&H" + A$)
40 NEXT I
50 DATA CD , OD , 1C , CD , 9E , CE , 21 ,
   08 , 01 , 22 , B8 , 00 , C9
60 DATA 3E , 08 , 32 , BB , 00 , 3E , 10 ,
   32 , BC , 00 , 3A , 10 , 00
70 DATA 32 , BD , 00 , C9
```

FIGURE 13 : EFFACEMENT

```
10 ' [
20 ' ORG $1C00
30 ' *** SCREEN ***
40 ' LD A . $00 : * SCREEN 1
50 ' LD ($D1) . A
60 ' LD C . A
70 ' LD A . $01
80 ' LD D . A : * PAGE ACTIVE - 1
90 ' LD E . A : * PAGE VISUELLE - 1
100 ' JP $AB09
110 ' ]
```

```
1C00 3E00        LD A , 00
1C02 32D100      LD (00D1) , A
1C05 4F          LD C , A
1C06 3201        LD A , 01
1C08 57          LD D , A
1C09 5F          LD E , A
1C0A C309AB      JP AB09
```

```
10 REM *** SCREEN ***
20 FOR I = &H1C00 TO &H1C0C
30 READ A$ : POKE I , VAL ("&H" + A$)
40 NEXT I
50 DATA 3E , 00 , 32 , D1 , 00 , 4F , 3E ,
   01 , 57
60 DATA 5F , C3 , 09 , AB
```

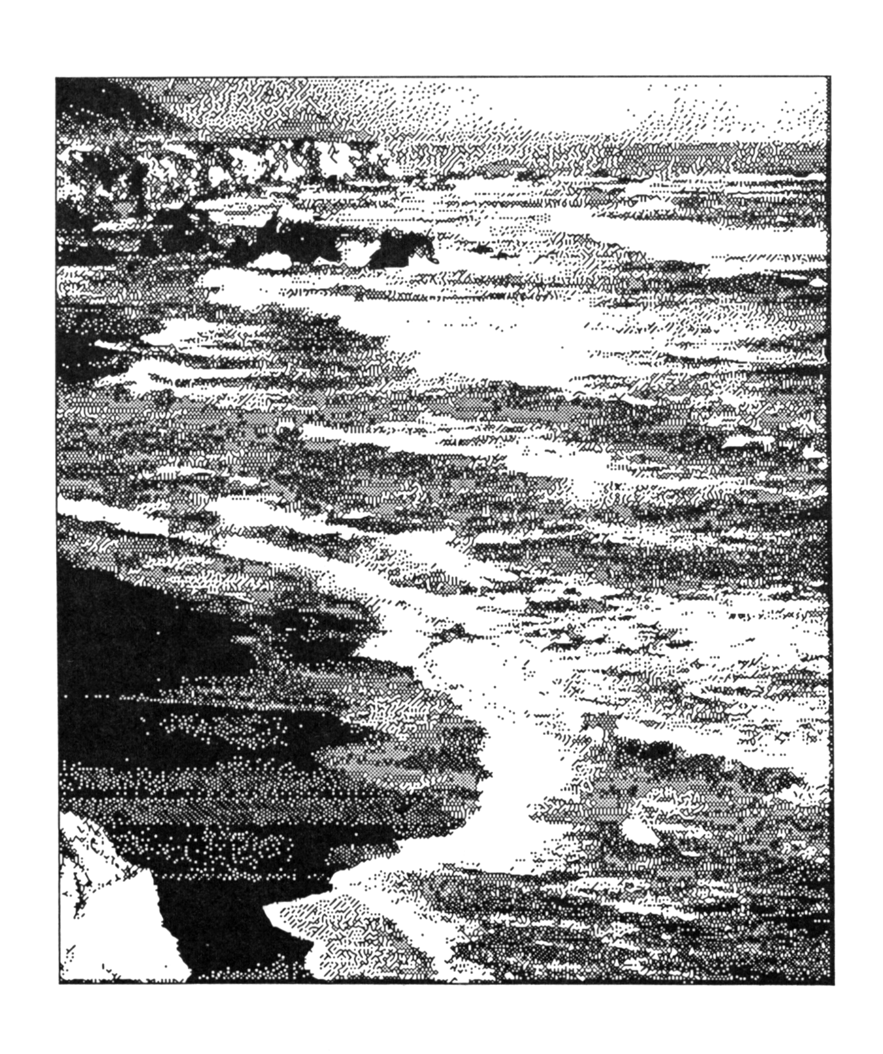
FIGURE 14 : SCREEN

```
10 ' [
20 ' ORG $1C00
30 ' *** COULEUR ***
40 ' LD A . $01
50 ' LD ($4E5) . A : * COULEUR LETTRES
60 ' LD A . $02
70 ' LD ($4E6) . A
80 ' LD A . $00
90 ' LD ($4E7) . A : * PALETTE 0 ou 2
100 ' RET
110 ' ]
```

```
1C00 3E01        LD A , 01
1C02 32E504      LD (04E5) , A
1C05 3E02        LD A , 02
1C07 32E604      LD (04E6) , A
1C0A 3E00        LD A , 00
1C0C 32E704      LD (04E7) , A
1C0F C9          RET
```

```
10 REM *** COLOR ***
20 FOR I = &H1C00 TO &H1C0F
30 READ A$ : POKE I , VAL ("&H" + A$)
40 NEXT I
50 DATA 3E , 01 , 32 , E5 , 04 , 3E , 02 ,
   32 , E6 , 04
60 DATA 3E , 00 , 32 , E7 , 04 , C9
```

FIGURE 15 : COLOR



EMBARQUONS-NOUS ...

CAP : "LES APPLICATIONS" !!!

3ème PARTIE :

... *Et où l'on termine "LOGICIEL" !*

INTRODUCTION

Enfin !! Nous voici arrivés à la dernière partie de cet ouvrage ... Et quelle partie !!

En effet , vous allez pouvoir vous régaler en savourant les délicieux logiciels que nous vous avons soigneusement préparés ... Des utilitaires aux jeux , en passant par la poésie , le menu est particulièrement appétissant !

Mais sans plus attendre , voici quelques indications avant de vous précipiter corps et âme sur les pages suivantes ...

Chaque logiciel est expliqué en détail : spécifications , principe , utilisation, résultats et listing sont présentés le plus clairement possible .

D'autre part , deux types de programmes sont exposés : ceux écrits exclusivement en BASIC et ceux composés en ASSEMBLEUR .

Pour ceux qui ne possèdent pas la K7 (Ils sont très courageux ...) , nous vous conseillons de relire plusieurs fois les codes entrés car la moindre erreur peut être fatale ! Un **"entreur de codes"** leur servira à entrer les codes en mémoire et un **"chargeur"** leur permettra de sauvegarder et de récupérer le logiciel via une K7 .

Evidemment , les personnes possédant la K7 n'ont pratiquement rien à faire ... L'ordre **CLOAD** est à utiliser avec les logiciels écrits uniquement en BASIC et l'ordre **CLOAD puis RUN** pour les logiciels ASSEMBLEUR . En effet , ces derniers possèdent un chargeur BASIC qui permet de stocker les codes rapidement en mémoire ... D'ailleurs un **RUN 100** permet de sauvegarder ces codes sur K7 afin de faire des copies protectrices .

Nous conseillons à toutes les personnes ne possèdant pas la K7 de se la procurer au plus vite auprès des **éditions NEPTUNE** ... En effet , en dehors de toute considération commerciale , il vaut mieux acquérir cette K7 car certains logiciels écrits en ASSEMBLEUR sont particulièrement longs ...

Voilà !! Nous espèrons que vous apprécierez les divers logiciels présents dans cette dernière partie : ils sont diversifiés et utilisent pleinement les notions abordées dans le présent ouvrage ... Analysez-les avec soin !

LMDATA & APPLICATIONS

La transcriptions en DATA d'une routine écrite en ASSEMBLEUR constitue une opération fastidieuse et génératrice d'erreurs ! Pourquoi ne pas confier ce travail à l'utilitaire "LMDATA" ? ...

L'utilisation de LMDATA est la suivante :

_ Implanter la routine L.M. (à transcrire en DATA) en mémoire .

_ Lancer LMDATA après l'avoir chargé dans le X-07 .

_ Supprimer les lignes de DATA superflues puis les lignes 1 à 230 . Cette opération peut se faire à l'aide de la carte XP-140 (commande DEL) , avec le fichier 5 , l'éditeur BASIC , etc ...

_ Si la routine LM accuse plus de 256 octets , il faut rajouter des lignes de DATA à la suite du logiciel initial . De plus , par mesure de prudence , ne pas exécuter LMDATA sans avoir effectué une copie K7 ou fichier RAM de la routine à implanter en DATA .

Le principe de LMDATA est le suivant :

Ligne 40 : calcul de l'adresse absolue du premier octet de la ligne 10000 .
Ligne 50 : entrée boucle des lignes complètes .
Ligne 60 : écriture de la ligne , calcul de l'adresse de la ligne suivante .
Ligne 70 : nombre de données des lignes incomplètes .
Ligne 200 : boucle des lignes de DATA .
Ligne 210 : calcul des variables C , D et U .
Ligne 220 : écriture des valeurs ASCII des variables C , D , U dans le programme puis ajustement de l'adresse mémoire .

Trois exemples sont joints à cet utilitaire : **le RESTORE calculé** que nous avons déjà vu en début d'ouvrage , **une copie rapide** de l'écran et un graphisme sur écran LCD représentant **le LOGO CANON** .

Longueur de LMDATA : 1670 octets .
Longueur du LOGO CANON : 831 octets .
Longueur du RESTORE calculé : 372 octets .
Longueur de COPIE RAPIDE : 539 octets .
Implantation de ces logiciels : BASIC .

```
1 REM Ecriture automatique de codes LM
 en DATA
10 CLEAR50,&H1F00:DEFINTA-Z:LL=16:CA=
48:CLS
20 INPUT"Adres. debut routine";DM:DD=DM
30 INPUT"Adresse fin routine";FM:NC=FM
 -DM+1:JC=INT(NC/LL)
40 RESTORE10000:AD=PEEK(&H328)+256*PEEK
(&H329)+6
50 LB=LL:FORJ=1TOJC
55 PRINT"LIGNE";10000+10*J
60 GOSUB200:AD=AD+5:NEXTJ
70 LB=NC-JC*LL:IFLB=0THEN100
80 GOSUB200
90 PRINT"Effacer la fin de la ligne";
 10000+10*JC
100 PRINT"Effacer les lignes";10000+10
*(JC+1);"";10150;
110 IFINKEY$=""THEN110
120 END
200 FORI=1TOLB:VC=PEEK(DM):DM=DM+1
210 C=INT(VC*.01):D=INT((VC-100*C)/10)
:U=VC-10*D-100*C
220 POKEAD,C+CA:POKEAD+1,D+CA:POKEAD+2
,U+CA:AD=AD+4
230 NEXT:RETURN
10000 DATA...,...,...,...,...,...,...,
 ...,...,...,...,...,...,...,...,...
10010 DATA...,...,...,...,...,...,...,
 ...,...,...,...,...,...,...,...,...
10020 DATA...,...,...,...,...,...,...,
 ...,...,...,...,...,...,...,...,...
10030 DATA...,...,...,...,...,...,...,.
.,...,...,...,...,...,...,...,...
10040 DATA...,...,...,...,...,...,...,
 ...,...,...,...,...,...,...,...,...
10050 DATA...,...,...,...,...,...,...,
 ...,...,...,...,...,...,...,...,...
10060 DATA...,...,...,...,...,...,...,
 ...,...,...,...,...,...,...,...,...
10070 DATA...,...,...,...,...,...,...,
 ...,...,...,...,...,...,...,...,...
10080 DATA...,...,...,...,...,...,...,
 ...,...,...,...,...,...,...,...,...
10090 DATA...,...,...,...,...,...,...,
 ...,...,...,...,...,...,...,...,...
10100 DATA...,...,...,...,...,...,...,
 ...,...,...,...,...,...,...,...,...
```

FIGURE 16 : LMDATA

```
10110 DATA...,...,...,...,...,...,...
 ...,...,...,...,...,...,...,...,...
10120 DATA...,...,...,...,...,...,...
 ...,...,...,...,...,...,...,...,...
10130 DATA...,...,...,...,...,...,...
 ...,...,...,...,...,...,...,...,...
10140 DATA...,...,...,...,...,...,...
 ...,...,...,...,...,...,...,...,...
10150 DATA...,...,...,...,...,...,...
 ...,...,...,...,...,...,...,...,...
```

```
1 REM Exemple de RESTORE CALCULE.
2 REM Cree par LMDATA
10 CLEAR50,&H1FFF:DEFINT A-Z
20 RESTORE10000
30 FORI=0TO17:READA:POKE&H2000+I,A:NE
40 INPUT"N= (1..3)";N
50 IFN<1 OR N>3 THEN 40
60 J=N*100
70 Z=USR(&H2000,J)
80 READA$:PRINTA$:GOTO40
100 DATA"lecture DATA 100"
200 DATA"lecture DATA 200"
300 DATA"lecture DATA 300"
999 END
10000 DATA035,035,094,035,086,205,013,
243,096,105,210,056,246,043,034,040
10010 DATA003,201
```

FIGURE 16 : FIN/FIGURE 17 : EX

```
REM EXEMPLE DE GRAPHISME

0 CLEAR200:DEFINTA-Z
0 X=23:CLS
0 FORL=6TO25
0 A1=1:F=1
0 READA:A1=A1+A:IFF=1THEN70
0 F=1:GOTO80
0 FORI=A1-ATOA1-1:PSET(X+I,L):NEXT:F=0
0 IFA1=<79THEN50
0 NEXT
00 END
10 DATA79,9,4,66,8,6,65,6,10,63
20 DATA5,12,7,4,9,3,3,2,7,7,6,3,3,2,6
30 DATA4,6,6,2,5,6,7,5,1,4,5,9,4,5,1,
4,5
40 DATA4,5,6,4,2,9,4,13,3,2,3,6,1,13,4
50 DATA3,6,6,2,10,3,6,5,2,5,2,3,2,6,3
,5,2,5,3
60 DATA3,5,6,1,12,4,5,5,2,5,1,4,3,6,2
,5,2,5,3
70 DATA3,5,15,8,5,5,2,5,1,4,3,6,2,5,2
,5,3
80 DATA3,5,14,10,4,5,2,5,1,5,3,5,2,5,
2,5,3
90 DATA3,5,13,5,2,4,4,5,2,5,1,5,3,5,
2,5,2,5,3
00 DATA3,6,11,5,4,4,3,5,2,5,1,6,3,4,
2,5,2,5,3
10 DATA4,5,9,1,1,5,4,4,3,5,2,5,1,6,3,
4,2,5,2,5,3
20 DATA4,6,6,2,2,6,2,6,2,5,2,5,2,6,2,
3,3,5,2,5,3
30 DATA5,12,4,8,1,4,2,5,2,5,2,6,3,2,
3,5,2,5,3
40 DATA6,10,6,6,2,5,1,5,2,5,3,9,4,5,
,5,3
50 DATA8,6,9,4,3,5,1,5,2,5,4,7,5,5,2
,5,3,79,79
```

FIGURE 18 : LOGO CANON

```
1 REM Copie cran
2 REM Mthode de la chaine
3 REM DATA cres par LMDATA.
10 CLEAR 100:DEFINT A-Z
20 CH$=STRING$(53,"#")
30 GOSUB 300
40 FORI=0TO52:READA:POKEAD+I,A:NEXT
100 PRINT" Copie rapide de l'"
110 PRINT"      ecran sur"
120 PRINT" l'imprimante X-710"
130 PRINT"          +++";
200 GOSUB300:EXEC AD
210 END
300 AD=VARPTR(CH$):AD=PEEK(AD+1)+256*
PEEK(AD+2):RETURN
10000 DATA006,000,033,020,002,197,229
001,019,000,009,126,254,032,032,004
10010 DATA043,013,032,247,012,225,229
126,004,035,229,197,205,247,206,193
10020 DATA225,120,185,032,242,205,176
207,225,001,020,000,009,193,004,120
10030 DATA254,004,032,209,201
```

FIGURE 19 : COPIE RAPIDE

LOGOGENESE

"Poètes , artistes , littérateurs , écrivains , admirez ce pur produit de la science informatique ... Adorez LOGOGENESE , le premier logiciel informatique qui va remplacer les académiciens !!"

Non , non , n'ayez pas peur ... Nous sommes en train de délirer mais cela va passer !! En restant sérieux , **LOGOGENESE est un programme de création de mots** : ils se forment au hasard à partir de syllabes stockées en DATA .

Evidemment , les mots ainsi créés ne vont pas avoir beaucoup de sens mais que voulez-vous , il faut bien dévoiler son originalité de temps en temps...

Le programme est très simple en lui-même . Ecrit entièrement en BASIC , il fait appel **aux méthodes du RESTORE calculé et de la chaîne de caractères** pour l'implantation d'une petite routine .

Pour l'utiliser , faire RUN suivi de RETURN dès que le programme se trouve en mémoire ... A partir de ce moment là , le X-07 crée des mots et les affiche à l'écran dans un bouillonnement de culture !!

Longueur de LOGOGENESE : 1819 octets .
Implantation de LOGOGENESE : BASIC .

```
'ATTENTION NE PAS RENUMEROTER
 DEFINTA-Z:CLS:PRINT"Logognse"
 RM$=STRING$(18,0):GOSUB90
 FORI=0TO17:READB$:POKEAD+I,VAL("&H"
$):NEXT:Z=RND(0)
 DATA23,23,5E,23,56,CD,0D,F3,60,69,
,38,F6,2B,22,28,03,C9
5 J=INT(RND(1)*33)+100:GOSUB90
 Z=USR(AD,J):N=INT(RND(1)*4+1)
5 FORI=1TON:READD$:NEXT
 K=INT(RND(1)*14)+200:GOSUB90
5 Z=USR(AD,K):N=INT(RND(1)*4+1)
 FORI=1TON:READF$:NEXT
5 PRINTD$;F$:IFINKEY$=""THEN45
 IFINKEY$=""THEN80ELSE45
 AD=VARPTR(RM$):AD=PEEK(AD+1)+256*
EK(AD+2):RETURN
5 END
```

FIGURE 20 : LOGOGENESE

```
100 DATA AERO,AGRO,AMBI,ANDRO
101 DATA ANTI,ASTRO,BIBLIO,BIO
102 DATA CACO,CALLI,CLEPTO,CHIMIO
103 DATA CHRONO,CINE,COPRO,COSMO
104 DATA CRYO,CRYPTO,DECI,DECA
105 DATA HEMI,DERMO,DEXTRO,DODECA
106 DATA DROMO,DYNAMO,DYS,ECTO
107 DATA ELECTRO,EMBRYO,ENTOMO
108 DATA ERGO,EROTICO,GASTRO,GEO
109 DATA GLOSSO,GONIO,GONO,GYMNO
110 DATA HECTO,HELIO,HEMATO,HETERO
111 DATA HIERO,HIPPO,HOLO,HOMEO
112 DATA HOMO,HYDRO,HYPER,HYPO
113 DATA ICONO,IDEO,INFRA,ISO
114 DATA INTRA,LATERO,LIPO,LOGO
115 DATA LOXO,MACRO,MEGALO,METEO
116 DATA META,]ICRO,MKsO,MNEMO
117 DATA MORPHO,MYTHO,NECRO,NEO
118 DATA NOSO,NUGLEO,OCTO,OLEO
119 DATA OMNI,ORTHO,PALEO,PAN
120 DATA PARA,PAPYRO,PATHO,PEDO
121 DATA PENTA,PERI,PETRO,PHAGO
122 DATA PHALLO,PHANERO,PHILO,PHYTO
123 DATA PHOBO,PHONO,PHOTO,PHRENO
124 DATA PHYSIO,POLY,MONO,HEMI
125 DATA PORNO,POST,PROTO,PSEUDO
126 DATA PSYCHO,RADIO,RETRO,RHINO
127 DATA SADO,STEREO,SCLERO,SEMI
128 DATA SCHIZO,SIMILI,SPELEO,STENO
129 DATA SUB,SUPRA,SUPER,TACHY
130 DATA TECHNO,TELE,TELEO,THEO
131 DATA THERMO,TRIBO,ULTRA,VIDEO
132 DATA XENO,XYLO,ZOO,MINI
200 DATA ALGIE,CARDE,CINESE,CEPHALE
201 DATA CLASTE,COSMOS,CYCLE,DACTYLE
202 DATA DERME,DIDACTE,DOXE,DROME
203 DATA DYNE,FUGE,GAME,GASTRE
204 DATA GLOTTE,GENESE,GENE,GRADE
205 DATA GYNE,LAIERE,LATRE,LOGIE
206 DATA LOGUE,MANCIE,MANE,MANIE
207 DATA MATIQUE,MEGALIE,METRIE,MNESIE
208 DATA MORPHE,NAUTE,NEVROSE,NOMIE
209 DATA PATHE,PHAGE,PHANIE,PHILE
210 DATA PHOBE,PHONE,PNEE,PYGE
211 DATA RRHEE,SCAPHE,SCOPE,SCOPIE
212 DATA STASE,STENIE,THEISME,TONIQUE
213 DATA TROPE,TYPE,VORE,TIQUE
```

FIGURE 20 : FIN

LLIST

Cet utilitaire est destiné à améliorer la présentation des listings BASIC : titre , numérotation des pages , saut de ligne en fin de page , alignements des débuts de ligne , formatage des numéros de ligne .

Le mode d'emploi de LLIST est le suivant :

_ Vérifier au moyen des commandes DIR et FSET qu'il reste au moins **600 octets libres** dans la zone des fichiers RAM . En effet , la partie active du logiciel va s'implanter automatiquement dans le fond de la mémoire .
_ Faire CLOAD puis RUN pour charger les codes si vous possédez la K7 des Editions NEPTUNE . Sinon , bon courage !!
_ Ajuster la longueur des pages (**octet &H950**) et la longueur des lignes (**octet &H93C**) à votre convenance . Pour ajuster ces octets , on peut utiliser la carte XP-140 ou la commande POKE . Remarquons que l'octet &H93C contient la valeur &H27 : il est préférable de la remplacer par &H26 pour éviter des sauts de ligne imprévus si une ligne de 39 caractères survient ...
_ Lancer le programme par **EXEC &H800** . La routine retourne alors à l'adresse décimale de son implantation : cette adresse est variable suivant votre version de X-07 . **Il est impératif de bien noter cette adresse afin de pouvoir lancer LLIST par "EXEC adresse" dès que l'on en a besoin.**

Le sous-programme relogeur , présenté dans les pages suivantes , s'auto-détruit automatiquement dès qu'il a exécuté sa mission .
LLIST utilise le même principe d'analyse d'un programme BASIC que les logiciels REFBAS ou EXABAS : vous pouvez donc vous y reporter afin de bien cerner le processus de travail .
D'autre part , la concision du programme résulte de l'emploi de la routine **&HFEFE** incluse dans la ROM . Cette routine réécrit dans le tampon d'entrée (début : &HD5) la chaîne pointée par le registre HL (terminateur : code 0) .

Longueur de LLIST : 554 octets .
Implantation de LLIST : variable (suivant la configuration) .

La routine permettant de reloger LLIST est décrite dans les lignes suivantes . Ce programme permet des quantités d'applications en contrôlant parfaitement l'implantation de n'importe quelle routine .

Ce "relocateur" a été réalisé avec un ASSEMBLEUR autorisant le calcul sur les labels symboliques . Si vous ne disposez pas d'un tel outil , vous devez , postérieurement à l'assemblage , calculer et recréer la table des adresses à modifier .

D'autre part , lors de la construction de la table des adresses à modifier , il faut être particulièrement vigilant **pour ne pas commettre un oubli irréparable** !!

Enfin , il est inutile d'écrire "un programme relogeur" si le logiciel à "relocater" n'est pas parfaitement au point ... En effet , toute modification du programme peut entrainer une ou des modifications de la table .

Voici le listing détaillé de cette routine :

```
DEB     LD HL , ($212)      ; HL contient l'adresse du fond de la
                              mémoire
        DEC HL
        LD BC , fin - début ; Longueur à déplacer
        XOR A               ; Mise du drapeau CARRY à 0
        SBC HL , BC         ; HL contient l'adresse du début du
                              programme
        PUSH HL             ;
        PUSH HL             ; Empilages pour usages ultérieurs
        PUSH HL             ;
        CALL $CE9E          ; Effacement de l'écran
        LD HL , MSG1        ; Affichage sur l'écran LCD du message :
        CALL $FEF7          ; "NOTEZ AD= ....."
        POP HL
        CALL $BB98          ; Affichage de AD en décimal
        LD HL , MSG2        ; Affichage sur l'écran LCD du message :
        CALL $FEF7          ; "Pressez une touche"
KEY     XOR A               ;
        CALL $C90A          ; Attente de la pression d'une touche
        JR Z , KEY          ;
        POP HL              ; HL contient le début après transfert
        LD DE , début       ; DE contient le début après transfert
        XOR A
        SBC HL , DE         ; HL = offset
        PUSH HL             ;
        POP BC              ; BC = offset
        LD IX , table       ; Table des adresses à modifier
```

```
DEB0    LD L , (IX+0)
        LD H , (IX+1)         ; HL = adresse à modifier
        LD E , (HL)
        INC HL
        LD D , (HL)           ; DE = contenu à modifier
        DEC HL                ; Réajustement de HL
        PUSH HL               ; Sauvegarde de l'adresse à modifier
        EX DE , HL            ; HL = contenu à modifier
        ADD HL , BC           ; HL = HL + offset
        EX DE , HL            ; DE = valeur modifiée
        POP HL                ; Récupération de l'adresse à modifier
        LD (HL) , E           ;
        INC HL                ; Modification effectuée
        LD (HL) , D           ;
        INC IX                ; Adresse suivante de la table
        INC IX                ;                  "
        LD A , (IX+0)
        CP 0                  ; Fini ?
        JR NZ , DEB0          ; Si ce n'est pas terminé , on boucle ...
        POP DE                ; ... Sinon DE = adresse début (MEM haute)
        LD HL , début         ; HL = adresse début (MEM basse)
        LD BC , fin - début   ; Longueur à transférer
        LDIR                  ; Transfert du programme modifié
        LD HL , DEB
        LD A , $C3            ; On écrit à la place de DEB les codes
        LD B , 3              ; "C3C3C3" afin de rendre le relocateur
DEB1    LD (HL) , A           ; inutilisable
        INC HL                ;
        DJNZ DEB1             ;
        JP $C3C3              ; Retour au BASIC
MSG1    DB 'NOTEZ AD =' , 0   ; 0 est le terminateur pour la routine FEF7
MSG2    DB 'Pressez ..." , 0  ; Message "Pressez une touche"
TABLE   DW A1+1 , .......
        DW ...........
        DB 0                  ; Fin de la table
```

Exemple de programme à reloger :

```
DEBUT   CALL $CE9E
A1      LD HL , MSIMP         ; Première adresse à modifier
        CALL $FEF7
        .........
FIN     EQU $                 ; Fin du logiciel à reloger
```

```
0800:2A 12 02 2B 01 6C 01 AF    *..+.l.
0808:ED 42 E5 E5 E5 CD 9E CE    mBeeeM.
0810:21 67 08 CD F7 FE E1 CD    ♀g.Mw~a
0818:98 BB 21 71 08 CD F7 FE    .;♀q.Mw
0820:AF CD 0A C9 28 FA E1 11    /M.I(za
0828:BD 08 AF ED 52 E5 C1 DD    =./mReA
0830:21 88 08 DD 6E 00 DD 66    ♀..]n.]
0838:01 5E 23 56 2B E5 EB 09    .^#V+ek
0840:EB E1 73 23 72 DD 23 DD    kas#r]#
0848:23 DD 7E 00 FE 00 20 E3    #]~.~.
0850:D1 21 BD 08 01 6C 01 ED    Q♀=..l.
0858:B0 21 00 08 3E C3 06 14    0♀..>C.
0860:77 23 10 FC C3 C3 C3 4E    w#.¦CCC
0868:6F 74 65 7A 20 41 44 3D    otez AD
0870:00 0D 0A 0A 50 72 65 73    ....Pre
0878:73 65 7A 20 75 6E 65 20    sez une
0880:74 6F 75 63 68 65 2E 00    touche.
0888:C1 08 D3 08 D6 08 E0 08    A.S.V.'
0890:E3 08 ED 08 FC 08 20 09    c.m.¦.
0898:32 09 37 09 3F 09 44 09    2.7.?.D
08A0:49 09 4D 09 53 09 58 09    I.M.S.X
08A8:5B 09 5E 09 61 09 66 09    [.^.a.f
08B0:6B 09 6F 09 72 09 7C 09    k.o.r.¦
08B8:BC 09 C1 09 00 CD 9E CE    <.A..M.
08C0:21 C5 09 CD F7 FE AF CD    ♀E.Mw~/
08C8:0A C9 28 FA CD 9E CE CD    .I(zM.N
08D0:B7 CF 21 04 0A 11 DF 09    7O♀..._
08D8:01 25 00 ED B0 3E 31 32    .%.m0>1
08E0:87 09 21 D5 09 CD F7 FE    ..♀U.Mw
08E8:CD F2 EB 23 11 DF 09 06    Mrk#._.
08F0:14 7E FE 00 28 05 12 23    .~~.(..
08F8:13 10 F6 CD 56 09 DD 2A    ..vMV.]
0900:B2 00 AF DD BE 01 CA C3    2./]>.J
0908:C3 DD E5 DD 6E 00 DD 66    C]e]n.]
0910:01 DD 5E 02 DD 56 03 E5    .]^.]V.
0918:DD E1 E1 01 04 00 09 CD    ]aa....
0920:9D 09 06 06 C5 CD FE FE    ....EM~
0928:C1 21 D5 00 7E FE 00 20    A♀U.~~.
0930:05 CD 43 09 18 CC CD 93    .MC..LM
0938:09 23 04 3E 27 B8 CC 43    .#.>'8L
0940:09 18 E9 CD 89 09 06 00    ..iM...
0948:3A 88 09 3C 32 88 09 FE    :..<2..
0950:38 C0 CD 56 09 C9 E5 21    8@MV.Ie
0958:81 09 CD 77 09 21 DF 09    ..Mw.♀_
0960:CD 77 09 3E 02 32 88 09    Mw.>.2.
0968:AF 47 3A 87 09 3C 32 87    /G:..<2
0970:09 21 FE 09 77 E1 C9 7E    .♀~.waI
0978:FE 00 C8 CD 93 09 23 18    ~.HM..#
```

FIGURE 21 : LLIST

```
0980:F6 0D 0A 0A 0A 0A 00 31    v
0988:02 E5 D5 C5 CD B0 CF C1    .
0990:D1 E1 C9 E5 D5 C5 CD F7    Q
0998:CE C1 D1 E1 C9 E5 EB 22    N
09A0:50 04 01 07 07 21 D5 00    P
09A8:CD 5F BE 21 D5 00 3E 30    M
09B0:BE 20 05 36 20 23 18 F6    >
09B8:21 D5 00 CD 77 09 3E 20    ♀
09C0:CD 93 09 E1 C9 49 6D 70    M
09C8:72 69 6D 61 6E 74 65 20    r
09D0:4F 4B 20 3F 00 54 69 74    O
09D8:72 65 20 3A 0A 0D 00 20    r
09E0:20 20 20 20 20 20 20 20
09E8:20 20 20 20 20 20 20 20
09F0:20 20 20 20 20 20 20 20
09F8:20 20 20 20 20 20 20 20
0A00:20 20 20 20 20 20 20 20
0A08:20 20 20 20 20 20 20 20
0A10:20 20 20 20 20 20 20 20
0A18:20 20 20 20 20 50 61 67
0A20:65 3A 20 31 20 0D 0A 0A    e
0A28:00 A5 2E 2E 2C 2E 2E 2E    .
```

FIGURE 21 : FIN

```
10 CLEAR50,&H7FF
20 INIT#1,"CASI:"
30 INPUT#1,N$,D,F
40 MOTOR
50 PRINT"Trouv :";N$
60 FORI=D-1TOF
70 POKEI,INP(#1)
80 NEXT:MOTOR
90 END
100 CLEAR50,&H7FF
110 D=&H800:F=&HA29
120 N$="LLIST":INIT#1,"CASO:"
130 INPUT"Magnto OK";T$
140 PRINT#1,N$,D,F:MOTOR
150 FORI=1TO1800:NEXT
160 FORI=DTOF:OUT#1,PEEK(I)
170 NEXT:MOTOR
180 END
```

```
5 '*** ENTREUR DE CODES ***
10 CLEAR50,&H7FF:A=&H800
20 PRINT HEX$(A);" : ";:INPUT C$
30 V=VAL ("&H"+C$):POKE A,V
49 A=A+1:IF A>&HA29 THEN PRINT "
...":BEEP 2,3:END
50 GOTO20
```

FIGURE 22 : CHARGEURS

```
LISTING LOGO CANON        Page: 1

   10 CLEAR200:DEFINTA-Z
   20 X=23:CLS
   30 FORL=6TO25
   40 A1=1:F=1
   50 READA:A1=A1+A:IFF=1THEN70
   60 F=1:GOTO80
   70 FORI=A1-ATOA1-1:PSET(X+I,L):NEXT
F=0
   80 IFA1=<79THEN50
   90 NEXT
  100 END
  110 DATA79,9,4,66,8,6,65,6,10,63
  120 DATA5,12,7,4,9,3,3,2,7,7,6,3,3,2
6
  130 DATA4,6,6,2,5,6,7,5,1,4,5,9,4,5,
,4,5
  140 DATA4,5,6,4,2,9,4,13,3,2,3,6,1,1
,4
  150 DATA3,6,6,2,10,3,6,5,2,5,2,3,2,6
3,5,2,5,3
  160 DATA3,5,6,1,12,4,5,5,2,5,1,4,3,6
2,5,2,5,3
  170 DATA3,5,15,8,5,5,2,5,1,4,3,6,2,5
2,5,3
  180 DATA3,5,14,10,4,5,2,5,1,5,3,5,2,
,2,5,3
  190 DATA3,5,13,5,2,4,4,5,2,5,1,5,3,5
2,5,2,5,3
  200 DATA3,6,11,5,4,4,3,5,2,5,1,6,3,4
2,5,2,5,3
  210 DATA4,5,9,1,1,5,4,4,3,5,2,5,1,6,
,4,2,5,2,5,3
  220 DATA4,6,6,2,2,6,2.6,2,5,2,5,2,6,
,3,3,5,2,5,3
  230 DATA5,12,4,8,1,4,2,5,2,5,2,6,3,2
3,5,2,5,3
  240 DATA6,10,6,6,2,5,1,5,2,5,3,9,4,5
2,5,3
  250 DATA8,6,9,4,3,5,1,5,2,5,4,7,5,5,
,5,3,79,79
```

FIGURE 23 : EXEMPLE

LABYRINTHE 3D

Ce jeu n'est pas unique en son genre mais il va vous faire passer d'agréables moments **grâce à sa rapidité et à ses multiples options** .

Le but est de se sortir d'un labyrinthe compliqué dont vous pouvez obtenir deux vues différentes : une vue de **"haut"** (totalité du dédale visible) ou une vue de **"bas"** (vue en perspective de la partie où vous vous trouvez ...) . La boussole affichée sur l'écran vous aidera à vous en sortir ... Du moins , nous l'espèrons !!

D'autre part , **un score est affiché** sur l'écran tout au long de la partie ... En effet , vous devez mettre le moins de temps possible à sortir du labyrinthe pour espérer obtenir le score le plus bas possible .

Le mode d'emploi est simple :

_ Après avoir chargé le logiciel (Soit CLOAD puis RUN , soit avec le chargeur de codes ...) , faire **EXEC &H800** .

_ Une petite présentation défile : tapez sur une touche pour continuer .

_ De succintes explications apparaissent accompagnées des commandes disponibles : **N**ord , **W**est , **E**st , **S**ud pour les direction , **B** pour retourner au BASIC , **R** pour repartir du point de départ , **A** pour avancer dans le dédale et **L** pour avoir une vue d'ensemble ... Cette dernière fonction coûte **20 points** : usez-en avec parcimonie !!

_ Après avoir tapé sur une touche quelconque , le labyrinthe se crée automatiquement ... Dès que vous êtes prêt à y pénétrer , pressez une touche ...

_ Vous vous trouvez alors à l'entrée du labyrinthe ... La sortie se trouve à l'opposé ! Rassurez-vous , le minotaure est en vacances !

Longueur de LABYRYNTHE : 2066 octets .
Implantation de LABYRINTHE : de 800h à 1012h .

Quelques indications sur **l'algorithme de création du dédale** :

1/ Créer un tableau de cellules vides .
2/ Choisir une origine voisine du centre .
3/ Tester les quatre cases voisines afin de déterminer celles qui ont été visitées .
4/ Si toutes les cases voisines ont été visitées , ajuster les pointeurs jusqu'à retomber dans une case déjà visitée , limitrophe d'une case non visitée .
5/ Choisir aléatoirement parmi les cellules limitrophes non visitées .
6/ Y aller en créant une porte dans la cellule de départ et une dans la cellule d'arrivée . Marquer la cellule d'arrivée .
7/ Si toutes les cellules du tableau n'ont pas été testées , remonter en 3/ sinon créer une entrée et une sortie .

```
0800:F3 21 CD 08 22 3D 00 FB s!M."=.{
0808:CD BD C0 CD 9E CE CD CB M=@M.NMK
0810:0D CD F3 08 2A 28 08 2B .Ms.*(.+
0818:22 28 08 7C B5 CA 8E 0A "(.¦5J..
0820:CD 4F 09 CD C7 09 18 EC MO.MG..l
0828:00 00 AF CD 0A C9 28 FA ../M.I(z
0830:C9 E5 D5 C5 FD E5 21 44 IeUE}e!D
0838:08 E5 FD 21 [illegible]7 00 21 00 .e}!..!.
0840:02 CD DF C2 2A 0E 00 7C .M_B*..¦
0848:B5 20 F9 21 59 08 E5 FD 5 y!Y.e}
0850:21 20 00 21 00 02 CD DF ! .!..M_
0858:C2 FD E1 C1 D1 E1 C9 D5 B}aAQaIU
0860:0E 20 CD 31 C2 D1 1A B7 . M1BQ.7
0868:C8 13 CD 8A C1 18 F7 ED H.M.A.wm
0870:5F 6F 3A C6 09 AD 17 6F _o:F.-.o
0878:ED 5F AD 32 C6 09 CD 83 m_-2F.M.
0880:08 7C C9 21 00 00 29 CB .¦I!..)K
0888:27 30 01 19 20 F8 C9 C5 '0.. xIE
0890:3E 14 01 00 04 CD 2F C9 >....M/I
0898:C1 C9 46 23 CD 8F 08 10 AIF#M...
08A0:FB C9 01 00 08 0B 78 B1 {I....x1
08A8:20 FB C9 22 50 04 21 A1  {I"P.!!
08B0:01 01 07 07 CD 5F BE 3E ....M_>>
08B8:30 21 A0 01 06 05 70 23 0! ...p#
08C0:BE 20 02 10 F9 EB FD E5 > ..yk}e
08C8:E1 CD 5F 08 C9 D9 08 DB aM_.IY.[
08D0:F1 FE 11 C2 9B C7 08 D9 q~.B.G.Y
08D8:ED 57 E2 C3 C3 CD BD C0 mWbCCM=@
08E0:3A 2B 00 06 08 B0 32 2B :+...02+
08E8:00 3E 0C D3 BB CD A2 00 .>.S;M".
08F0:C3 CF C3 CD 9E CE 21 40 COCM.N!@
08F8:01 22 28 08 E5 C1 21 12 ."(.eA!.
0900:10 11 13 10 36 00 ED B0 ....6.m0
0908:01 03 21 21 70 02 3E FE ..!!p.>~
0910:77 2B 3E 01 77 23 23 3E w+>.w##>
0918:1F 77 21 70 02 7E 81 77 .w!p.~.w
0920:2B 2B 77 CD 8F 08 10 F2 ++wM...r
0928:01 03 0B 21 6F 02 3E FE ...!o.>~
0930:77 2B 3E 01 77 23 23 3E w+>.w##>
0938:61 77 21 6F 02 7E 81 77 aw!o.~.w
0940:23 23 77 21 6E 02 CD 8F ##w!n.M.
0948:08 10 EF 11 05 10 C9 62 ..o...Ib
0950:6B 0E 00 CD 61 0A DD 22 k..Ma.]"
0958:C4 09 7B FE 00 28 0D 1D D.{~.(..
0960:CD 61 0A 5D 20 06 CB C1 Ma.] .KA
0968:DD 22 BC 09 7B FE 09 28 ]"<.{~.(
0970:0D 1C CD 61 0A 5D 20 06 ..Ma.] .
0978:CB D1 DD 22 BE 09 7A FE KQ]">.z~
0980:00 28 0D 15 CD 61 0A 54 .(..Ma.T
```

FIGURE 24 : LABYRINTHE 3D

```
0988:20 06 CB D9 DD 22 C0 09
0990:7A FE 1F 28 0D 14 CD 61
0998:0A 54 20 06 CB C9 DD 22
09A0:C2 09 79 B7 C0 14 3E 1F
09A8:BA 30 0A 16 00 1C 3E 09
09B0:BB 30 02 1E 00 CD 61 0A
09B8:28 EB 18 93 0B 11 47 11
09C0:4A 11 0C 11 27 11 ED CD
09C8:4E 0A 2A C4 09 F5 86 77
09D0:F1 CB 4F 20 47 CB 57 20
09D8:18 CB 5F 20 2B 2A BC 09
09E0:3E 04 CD 40 0A 1D 04 70
09E8:23 71 04 23 70 23 71 18
09F0:40 2A BE 09 3E 01 CD 40
09F8:0A 1C 3E 03 81 04 70 23
0A00:77 04 23 70 23 77 18 29
0A08:2A C0 09 3E 02 CD 40 0A
0A10:15 70 0C 23 71 23 70 23
0A18:0C 71 18 15 2A C2 09 3E
0A20:08 CD 40 0A 14 3E 03 80
0A28:77 23 0C 71 23 77 23 0C
0A30:71 21 6E 02 CD 37 0A 3E
0A38:12 01 00 02 CD 2F C9 C9
0A40:86 77 D5 E1 29 19 24 2C
0A48:E5 C1 21 6E 02 C9 D5 11
0A50:04 00 CD 6F 08 3C FE 03
0A58:20 02 3E 08 A1 28 F0 D1
0A60:C9 E5 C5 26 00 44 6B 4A
0A68:29 29 29 29 29 09 01 12
0A70:10 09 3E 00 BE E5 DD E1
0A78:C1 E1 C9 31 2F 34 32 2F
0A80:30 39 2F 37 32 2F 31 38
0A88:31 2F 30 38 39 00 CD 31
0A90:08 06 1D 3E 61 CD 87 0F
0A98:21 AD 03 CB CE CD 2A 08
0AA0:21 00 00 22 ED 0C 11 00
0AA8:00 3E 03 32 EF 0C CD 10
0AB0:0C CD 2A 08 FE 42 CA C0
0AB8:0D FE 4E 20 04 3E 01 18
0AC0:EA FE 45 20 04 3E 02 18
0AC8:E2 FE 57 20 04 3E 04 18
0AD0:DA FE 53 28 D4 FE 41 28
0AD8:1A FE 52 28 B1 FE 4C 20
0AE0:D0 CD A7 0F 2A ED 0C 01
0AE8:14 00 09 22 ED 0C CD 2A
0AF0:08 18 B6 CD 4B 0B 2A ED
0AF8:0C 23 22 ED 0C 3A F0 0C
0B00:CB 47 CA B1 0A 3A EF 0C
0B08:FE 01 20 03 1D 18 0F FE
0B10:02 20 03 14 18 08 FE 03
```

FIGURE 24 : SUITE

```
0B18:20 03 1C 18 01 15 21 09 ......!.
0B20:1F E7 C2 AE 0A CD 31 08 .gB..M1.
0B28:CD 9E CE 21 97 0E CD F7 M.N!..Mw
0B30:FE FD 21 02 02 2A ED 0C ~}!..*m.
0B38:CD AB 08 CD 2A 08 FE 31 M+.M*.~1
0B40:CA A0 0A FE 32 CA 11 08 J .~2J..
0B48:C3 C0 0C CD 61 0A D5 21 C@.Ma.U!
0B50:F0 0C 3A EF 0C 06 03 FE p.:o...~
0B58:04 28 20 FE 02 28 30 FE .( ~.(0~
0B60:03 28 40 DD 7E 00 77 23 .(@]~.w#
0B68:7B FE 00 28 0A 1D C5 01 {~.(..E.
0B70:E0 FF DD 09 C1 10 EC 36 `.].A.l6
0B78:FF D1 C9 DD 7E 00 CD BC .QI]~.M<
0B80:0B 77 23 7A FE 00 28 EF .w#z~.(o
0B88:15 CD 2B 10 EE 18 E8 DD .M+.n.h]
0B90:7E 00 CD F4 0B 77 23 7A ~.Mt.w#z
0B98:FE 1F 28 DB 14 DD 23 10 ~.([.]#.
0BA0:EE 18 D4 DD 7E 00 CD D8 n.T]~.MX
0BA8:0B 77 23 7B FE 09 28 C7 .w#{~.(G
0BB0:1C C5 01 20 00 DD 09 C1 .E. .].A
0BB8:10 E9 18 BB 0E 00 CB 5F .i.;..K_
0BC0:28 02 CB C1 CB 57 28 02 (.KAKW(.
0BC8:CB D9 CB 4F 28 02 CB D1 KYKO(.KQ
0BD0:CB 47 28 02 CB C9 79 C9 KG(.KIyI
0BD8:0E 00 CB 5F 28 02 CB C9 ..K_(.KI
0BE0:CB 57 28 02 CB C1 CB 4F KW(.KAKO
0BE8:28 02 CB D9 CB 47 28 02 (.KYKG(.
0BF0:CB D1 79 C9 0E 00 CB 5F KQyI..K_
0BF8:28 02 CB D1 CB 57 28 02 (.KQKW(.
0C00:CB C9 CB 4F 28 02 CB C1 KIKO(.KA
0C08:CB 47 28 02 CB D9 79 C9 KG(.KYyI
0C10:D5 CD 9E CE CD 4B 0B 21 UM.NMK.!
0C18:F5 0C CD 9A 08 3A EF 0C u.M..:o.
0C20:47 21 0A 0D 11 08 00 19 G!......
0C28:10 FD EB 21 04 04 CD 5F .}k!..M_
0C30:08 21 01 0F 11 E1 0E CD .!...a.M
0C38:5F 08 FD 21 02 10 2A ED _.}!..*m
0C40:0C CD AB 08 11 F0 0C 1A .M+..p..
0C48:CB 4F 20 08 21 57 0D CD KO .!W.M
0C50:9A 08 18 06 21 44 0D CD ....!D.M
0C58:9A 08 1A CB 5F 20 08 21 ...K_ .!
0C60:52 0D CD 9A 08 18 06 21 R.M....!
0C68:3F 0D CD 9A 08 1A CB 47 ?.M...KG
0C70:20 25 21 32 0D 18 6C 21  .!2..l!
0C78:49 0D CD 9A 08 13 1A CB I.M....K
0C80:4F 20 08 21 89 0D CD 9A O .!..M.
0C88:08 18 06 21 72 0D CD 9A ...!r.M.
0C90:08 1A CB 5F 20 08 21 84 ..K_ .!.
0C98:0D CD 9A 08 18 06 21 69 .M....!i
0CA0:0D CD 9A 08 1A CB 47 20 .M...KG
```

FIGURE 24 : SUITE

```
0CA8:05 21 5C 0D 18 35 21 7B .!\..5!{
0CB0:0D CD 9A 08 13 1A CB 4F .M....KO
0CB8:20 08 21 BB 0D CD 9A 08  .!;.M.
0CC0:18 06 21 A4 0D CD 9A 08 ..!$.M.
0CC8:1A CB 5F 20 08 21 B6 0D .K_ .!6
0CD0:CD 9A 08 18 06 21 9B 0D M....!.
0CD8:CD 9A 08 1A CB 47 20 08 M...KG
0CE0:21 8E 0D CD 9A 08 D1 C9 !..M..Q
0CE8:21 AD 0D 18 F6 00 00 03 !-..v..
0CF0:09 0D 05 FF 00 07 0D 17 .......
0CF8:00 17 00 16 00 00 01 00 .......
0D00:4D 00 4D 01 4D 17 4C 17 M.M.M.L
0D08:40 17 0D 16 0D 01 40 01 @.....@
0D10:40 17 57 20 20 4E 20 20 @.W  N
0D18:45 00 4E 20 20 45 20 20 E.N  E
0D20:53 00 45 20 20 53 20 20 S.E  S
0D28:57 00 53 20 20 57 20 20 W.S  W
0D30:4E 00 03 11 14 3C 14 10 N....<.
0D38:14 10 01 3D 14 3D 01 01 ...=.=.
0D40:10 14 10 01 01 3D 14 3D .....=.
0D48:01 02 11 14 15 10 38 10 ......8
0D50:3C 14 01 0E 17 10 15 01 <......
0D58:3F 17 3D 15 03 19 0C 34 ?.=....
0D60:0C 18 0C 18 01 35 0C 35 .....5.
0D68:01 02 15 0F 15 01 18 01 .......
0D70:18 0C 02 35 0C 35 01 38 ...5.5.
0D78:21 38 0F 02 19 0C 1D 08 .8.....
0D80:30 08 34 0C 01 16 0F 18 0.4....
0D88:0D 21 35 0D 37 0F 03 21 ..5.7..
0D90:04 2C 04 20 01 20 04 2D .,. . .
0D98:01 2D 04 02 1D 07 1D 01 .-.....
0DA0:20 01 20 04 02 2D 04 2D  . ..-.
0DA8:01 30 01 30 07 02 21 04 .0.0..!
0DB0:24 01 29 01 2C 04 01 1E $.).,..
0DB8:07 20 05 01 2D 05 2F 07 . ..-./
0DC0:F3 21 39 C7 22 3D 00 FB s!.G"=.
0DC8:C3 28 C5 CD 9E CE CD 31 C(EM.NM
0DD0:08 CD BD C0 21 87 0E CD .M=@!..M
0DD8:3A 08 21 04 08 22 B8 0D :.!.."8
0DE0:21 90 0E CD F7 FE 0E 0E !..Mw~.
0DE8:2E 02 CD 0A C9 C7 4E 0E ..M.IGN.
0DF0:C5 01 00 35 CD A5 08 C1 E..5M%.A
0DF8:79 FE 0D 38 0D CD 35 0E y~.8.M5.
0E00:0C 79 FE 15 20 E4 0E 01 .y~. d..
0E08:18 E0 C5 3E 15 91 67 41 .`E>..gA
0E10:11 79 0E 1A 4F D5 C5 CD .y..OUEM
0E18:31 C2 C1 D1 13 24 10 F3 1BAQ.$.s
0E20:26 21 1A FE 00 28 0A 4F &..~.(.O
0E28:D5 CD 31 C2 D1 13 24 18 UM1BQ.$.
0E30:F1 C1 0C 18 B5 C5 3E 15 qA..5E>.
```

FIGURE 24 : SUITE

```
38:91 67 11 79 0E 1A 4F FE   .g.y..C~
40:00 28 09 D5 CD 31 C2 D1   .(.UM1BQ
48:13 24 18 F1 C1 C9 CD 9E   .$.qAIM.
50:CE 21 E7 0E CD F7 FE CD   N!g.Mw~M
58:2A 08 CD 9E CE 21 30 0F   *.M.N!0.
60:CD F7 FE CD 2A 08 CD 9E   Mw~M*.M.
68:CE 21 02 05 11 7E 0F CD   N!...~.M
70:5F 08 01 00 00 CD A5 08   _....M%.
78:C9 2A 4C 41 42 59 52 49   I*LABYRI
80:4E 54 48 45 2A 20 00 02   NTHE* ..
88:00 05 77 05 00 11 77 11   ..w...w.
90:84 85 84 85 84 85 00 42   .......B
98:52 41 56 4F 20 56 6F 75   RAVO Vou
A0:73 20 73 6F 72 74 65 7A   s sortez
A8:20 65 6E 20 20 20 20 20    en
B0:20 20 63 6F 75 70 73 2E     coups.
B8:0D 0A 31 2D 6C 65 20 6D   ..1-le m
C0:9B 6D 65 20 6C 61 62 79   .me laby
C8:72 69 6E 74 68 65 32 2D   rinthe2-
D0:75 6E 20 61 75 74 72 65   un autre
D8:20 20 33 2D 66 69 6E 2E     3-fin.
E0:00 53 63 6F 72 65 00 56   .Score.V
E8:6F 75 73 20 64 65 76 65   ous deve
F0:7A 20 61 74 74 65 69 6E   z attein
F8:64 72 65 6C 61 20 73 6F   drela so
00:72 74 69 65 20 65 6E 20   rtie en
08:73 75 69 76 61 6E 74 6C   suivantl
10:65 73 20 69 6E 64 69 63   es indic
18:61 74 69 6F 6E 73 20 64   ations d
20:65 20 20 6C 61 20 62 6F   e  la bo
28:75 73 73 6F 6C 65 2E 00   ussole..
30:4E 53 45 57 3D 54 6F 75   NSEW=Tou
38:72 6E 65 2F 41 3D 41 76   rne/A=Av
40:61 6E 63 65 52 3D 52 65   anceR=Re
48:63 6F 6D 6D 65 6E 63 65   commence
50:2F 42 3D 42 61 73 69 63   /B=Basic
58:4C 3D 56 75 65 20 64 75   L=Vue du
60:20 6C 61 62 79 72 69 6E    labyrin
68:74 68 65 20 28 63 6F 96   the (co.
70:74 20 32 30 20 70 6F 69   t 20 poi
78:6E 74 73 29 2E 00 43 72   nts)..Cr
80:9C 61 74 69 6F 6E 00 21   .ation.!
88:6E 02 77 23 70 23 77 04   n.w#p#w.
90:23 70 CD 31 0A C9 04 01   #pM1.I..
98:01 61 01 61 01 61 1F 61   .a.a.a.a
A0:1F 01 1F 01 1F 01 01 D5   .......U
A8:CD 9E CE 21 96 0F CD 9A   M.N!..M.
B0:08 21 12 10 1E 00 16 00   .!......
B8:CB 4E 20 03 CD E9 0F CB   KN .Mi.K
```

FIGURE 24 : FIN

```
0FC0:56 20 03 CD 02 10 23 14   V .M.
0FC8:3E 20 BA 20 EB 1C 3E 0A   > : k
0FD0:BB 20 E3 3E 61 06 1D CD   ; c>a
0FD8:87 0F D1 CD 42 0A 04 0C   ..QMB
0FE0:70 23 71 04 0C 23 E5 18   p#q..
0FE8:0E E5 CD 42 0A 04 04 04   .eMB.
0FF0:70 23 71 23 0C 0C 0C 70   p#q#.
0FF8:23 71 21 6E 02 CD 8F 08   #q!n.
1000:E1 C9 E5 CD 42 0A 0C 0C   aIeMB
1008:0C 70 23 71 23 04 04 04   .p#q#
1010:18 E5 06 08 06 0E 0A 0C   .e...

2 '*** PROGRAMME 1 ***
5 '*** CHARGEUR BASIC ***
10 CLEAR50,&H7F0
20 INIT#1,"CASI:"
30 INPUT#1,N$,D,F
40 MOTOR
50 PRINT"Trouv :";N$
60 FORI=D-1TOF
70 POKEI,INP(#1)
80 NEXT:MOTOR
90 END
100 CLEAR50,&H7F0
110 D=&H800:F=&H1012
120 N$="PLABIR":INIT#1,"CASO:"
130 INPUT"Magnto OK";T$
140 PRINT#1,N$,D,F:MOTOR
150 FORI=1TO1800:NEXT
160 FORI=DTOF:OUT#1,PEEK(I)
170 NEXT:MOTOR
180 END

2 '*** PROGRAMME 2 ***
5 '*** ENTREUR DE CODES ***
10 CLEAR50,&H7F0:A=&H800
20 PRINT HEX$(A);" : ";:INPUT C$
30 V=VAL ("&H"+C$):POKE A,V
49 A=A+1:IF A>&H1012 THEN PRINT "
...":BEEP 2,3:END
50 GOTO20
```

FIGURE 25 : CHARGEURS

LE SOLITAIRE

Pas un jeu n'est à la fois plus familier et plus méconnu que **le SOLITAIRE** . Ses règles sont si simples qu'il n'a pas la réputation d'être un jeu de réflexion . Néanmoins , **il exclut toute forme de hasard** , ce qui en fait un excellent casse-tête .

Le matériel qu'il requiert est des plus simples : il se réduit à une planchette perçée de **33 , 37 ou 41 trous** dans lesquels viennent se loger autant de fiches , billes ou coquillages . Le solitaire à 33 trous , aussi appelé **"Solitaire ANGLAIS"** , connut un grand succès en Angleterre et en Allemagne . Le logiciel proposé utilise d'ailleurs le type anglais . Notons tout de même que deux autres versions existent : le **"solitaire FRANCAIS"** à 37 cases et le **"GRAND Solitaire"** à 41 trous .

Les règles du jeu sont des plus faciles à assimiler . Il suffit de mettre en place toutes les fiches dans les trous prévus à cet effet , d'en ôter une au choix puis de commencer à les retirer comme on le fait au jeu de DAMES . Remarquons qu'une fiche peut prendre une autre fiche contiguë en sautant par dessus , horizontalement ou verticalement , à condition de retomber dans un trou inoccupé .

Si aucun mouvement ne peut être effectué , la partie est terminée ... Plus le nombre de fiches est restreint sur le damier , meilleur est le score , l'idéal étant qu'un seule fiche demeure .

Le mode d'emploi est des plus simples :

_ Après avoir chargé le programme dans la mémoire du X-07 par CLOAD puis RUN (ou par le chargeur de codes si vous ne possédez pas la K7 ...) , faire **EXEC &H800** .

_ **L'option 1** du menu permet d'accéder à **une recherche par le X-07.** L'utilisateur choisit le nombre de pions de la solution finale et le programme livrera une solution après un laps de temps plus ou moins long , suivant l'option choisie . En effet , le canoniste a le choix entre 22 possibilités de recherche .

Aucun algorithme simple n'est connu pour ce jeu et le programme effectue **une recherche systématique** . A chaque étape , il examine successivement toutes les cases dans les quatre directions possibles ... Dès qu'un coup est jouable , le X-07 l'exécute . Si la recherche conduit à une impasse , le CANON remonte niveau par niveau : l'ordre d'examen des directions possède donc une importance capitale en ce qui concerne la vélocité avec laquelle le logiciel parvient à une solution .

_ **L'option 2** du menu donne accès au **jeu du joueur** : le programme assure l'affichage du damier , enregistre , teste et exécute les coups proposés par le joueur . Plusieurs commandes sont disponibles pour jouer :

* Déplacer le curseur sur la case de départ au moyen des flèches et presser sur la touche **"D"** : un marqueur s'allumera ...

* Déplacer ensuite le curseur sur la case d'arrivée et presser la touche **"A"** afin d'exécuter le coup .

* La touche **"-"** permet de revenir au coup précédent .

* La touche **"Q"** permet d'arrêter le jeu quand il est bloqué .

* La commande **"CTRL Q"** redonne "la main" au BASIC .

_ Si le nombre de pions est **supérieur ou égal à 3** dans l'option 1 , le logiciel trouve une réponse en **quelques secondes** . Par contre , si le nombre de pions est **égal à 2** , l'attente peut varier de **cinq minutes à une heure** selon le choix de l'ordre . Pour l'ordre 9 , le programme trouve une solution classique en moins de **dix minutes** .

_ Enfin , le logiciel affiche la profondeur d'analyse dans l'arbre . Cette opération se révèle très couteuse en temps d'exécution ... Pour la supprimer , il suffit de mettre à **0** les octets &H989 , &H98A et &H98B par l'ordre POKE du BASIC , par exemple .

_ Voici , pour terminer , **un schéma du Solitaire anglais** :

		1	2	3		
		4	5	6		
7	8	9	10	11	12	13
14	15	16	17	18	19	20
21	22	23	24	25	26	27
		28	29	30		
		31	32	33		

<u>Longueur de SOLITAIRE</u> : 4609 octets .

<u>Implantation de SOLITAIRE</u> : de 800h à 11D5h .

```
0800:3E 33 CD 28 E4 F3 21 4D   >3M(dsºM
0808:0D 22 3D 00 FB CD BD C0   ."=.{M=@
0810:CD 9E CE CD 05 0D 06 04   M.NM....
0818:21 AD 0D CD 34 0D 21 02   º-.M4.º.
0820:06 11 D7 0D CD 24 0D 21   ..W.M$.º
0828:03 06 11 55 0E CD 24 0D   ...U.M$.
0830:21 71 11 AF 77 11 72 11   ºq./w.r.
0838:01 64 08 ED B0 CD F2 0C   .d.m0Mr.
0840:CD 9E CE CD 05 0D 21 D4   M.NM..ºT
0848:11 3E 01 77 11 D5 11 01   .>.w.U..
0850:21 00 ED B0 21 E4 11 AF   º.m0ºd./
0858:77 21 E1 0D CD F7 FE CD   wºa.Mw~M
0860:F2 0C FE 31 28 07 FE 32   r.~1(.~2
0868:CA 84 0B 18 F2 CD 9E CE   J...rM.N
0870:21 85 0E CD F9 0C FE 21   º..My.~º
0878:30 F3 3E 21 93 32 B0 10   0s>º.20.
0880:CD 9E CE 21 39 0E CD F9   M.Nº9.My
0888:0C FE 17 30 E0 3D 87 87   .~.0`=..
0890:87 26 00 6F 11 B1 10 19   .&.o.1..
0898:11 2C 09 CD B3 08 11 33   .,.M3..3
08A0:09 CD B3 08 11 3A 09 CD   .M3..:.M
08A8:B3 08 11 41 09 CD B3 08   3..A.M3.
08B0:C3 CE 08 7E 12 23 13 7E   CN.~.#.~
08B8:12 23 C9 7F FF FF 7F 7F   .#I.....
08C0:00 C5 06 00 CB 21 DD 21   .E..Kº]º
08C8:90 11 DD 09 C1 C9 CD 9E   ..].AIM.
08D0:CE 21 02 04 11 C7 0E CD   Nº...G.M
08D8:24 0D 21 D4 11 22 72 09   $.ºT."r.
08E0:11 01 00 01 01 00 04 3E   .......>
08E8:05 B8 28 07 AF B9 CA CB   .8(./9JK
08F0:0A 18 28 1C 06 01 3E 22   ..(...>"
08F8:BB 20 F1 CD C1 08 DD 36   ; qMA.]6
0900:00 00 DD 36 01 00 0D 2A   ..]6...*
0908:72 09 11 40 00 AF ED 52   r..@./mR
0910:22 72 09 DD 5E FE DD 46   "r.]^~]F
0918:FF 18 CB C5 D5 7B 32 6F   ..KEU{2o
0920:09 2A 72 09 19 2B 7E B7   .*r..+~7
0928:28 40 78 21 6C 10 FE 01   (@xºl.~.
0930:28 11 21 E4 0F FE 02 28   (.ºd.~.(
0938:0A 21 28 10 FE 03 28 03   .º(.~.(.
0940:21 A0 0F CB 23 19 7E B7   º..K#.~7
0948:28 20 32 70 09 5F 23 FD   ( 2p._#}
0950:2A 72 09 FD 19 FD 7E FF   *r.}.}~.
0958:B7 20 0F 5E 7B 32 71 09   7 .^{2q.
0960:2A 72 09 19 2B 7E FE 01   *r..+~~.
0968:28 0A D1 C1 C3 E6 08 21   (.QACf.º
0970:19 1E 14 19 D1 C1 CD C1   ....QAMA
0978:08 DD 73 00 DD 70 01 0C   .]s.]p..
0980:C5 D5 3E 30 81 4F 21 02   EU>0.Oº.
0988:13 00 00 00 D1 C1 06 00   ....QA..
```

FIGURE 26 : SOLITAIRE

```
0990:1E 01 CD A5 09 3A B0 10   ..M%.
0998:B9 C2 E6 08 CD E9 09 CC   9Bf.M
09A0:F4 0A C3 FB 08 C5 D5 11   t.C{.
09A8:40 00 21 94 11 41 19 10   @.º..
09B0:FD 22 72 09 E5 AF ED 52   }"r.e
09B8:D1 D5 01 21 00 ED B0 D1   QU.º.
09C0:D5 FD E1 1B 26 00 3A 6F   U}a.&
09C8:09 6F FD 77 21 19 AF 77   .o}wº
09D0:26 00 3A 70 09 6F FD 77   &.:p.
09D8:22 19 3E 01 77 26 00 3A   ".>.w
09E0:71 09 6F 19 AF 77 D1 C1   q.o./
09E8:C9 C5 2A 72 09 06 21 AF   IE*r.
09F0:86 23 10 FC 4F 3A B0 10   .#.|O
09F8:81 FE 21 C1 C9 00 00 00   .~ºAI
0A00:FF FF FF FF FF FF FF FF   .....
0A08:FF FF FF FF FF FF FF FF   .....
0A10:FF FF FF FF 7F FF FF FF   .....
0A18:FF FF 7F 7F FF FF FF FF   .....
0A20:7F FF FF FF FF FF FF FF   .....
0A28:FF FF FF FF 7F 7F 7F 7F   .....
0A30:06 10 21 1E 0F CD 34 0D   ..º..
0A38:2A 72 09 11 01 00 7E FE   *r...
0A40:01 28 0F E5 D5 CD BD 0A   .(.eU
0A48:D1 E1 23 13 7B FE 22 20   Qa#.{
0A50:ED C9 E5 D5 CD B9 0A D1   mIeUM
0A58:E1 18 EF CD BD C0 3E 82   a.oM=
0A60:01 01 00 D5 11 C4 01 CD   ...U.
0A68:2F C9 1A D1 FE 33 28 13   /I.Q~
0A70:FE 37 28 1B FE 35 28 0B   ~7(.~
0A78:FE 31 28 01 C9 21 A0 0F   ~1(.I
0A80:E5 18 10 21 28 10 E5 18   e..º(
0A88:0A 21 6C 10 E5 18 04 21   .ºl.e
0A90:E4 0F E5 CD AC 0A E1 19   d.eM,
0A98:23 5E AF C9 11 00 3E 13   #^/I.
0AA0:01 20 02 CD 2F C9 01 00   ...M/
0AA8:30 C3 47 [illegible] ED 5B 9C 0A   0CG.m
0AB0:2A 72 09 2B 19 7E B7 28   *r.+.
0AB8:04 3E 11 18 02 3E 12 CB   .>...
0AC0:23 21 5C 0F 19 01 00 02   #º\..
0AC8:C3 2F C9 CD 9E CE CD 05   C/IM.
0AD0:0D 21 02 02 11 EB 0E CD   .º...
0AD8:24 0D CD F2 0C CD 9E CE   $.Mr.
0AE0:21 D7 0E CD F7 FE CD 05   ºW.Mw
0AE8:0D CD F2 0C FE 4E CA BD   .Mr.~
0AF0:0D C3 10 08 CD 9E CE 21   .C..M
0AF8:FB 0E CD F7 FE CD 05 0D   {.Mw~
0B00:CD F2 0C CD 9E CE 21 D4   Mr.M.
0B08:11 22 72 09 CD 30 0A 0E   ."r.M
0B10:2D 21 02 0D CD 31 C2 CD   -º..M
0B18:F2 0C 3A B0 10 3D 00 47   r.:0.
```

FIGURE 26 : SUITE

```
0B20:21 14 12 11 40 00 E5 C5   !...@.eE
0B28:D5 E5 FD E1 22 72 09 CD   Ue}a"r.M
0B30:3B 0A FD 6E 21 26 00 DD   ;.}n!&.]
0B38:21 02 0A CD 73 0D FD 6E   !..Ms.}n
0B40:22 26 00 DD 21 02 0F CD   "&.]!..M
0B48:73 0D CD F2 0C D1 C1 E1   s.Mr.QAa
0B50:19 10 D3 CD F2 0C 21 D7   ..SMr.!W
0B58:0E CD F7 FE CD F2 0C FE   .Mw~Mr.~
0B60:4E CA BD 0D CD 9E CE 21   NJ=.M.N!
0B68:CA 0E CD F7 FE CD F9 0C   J.Mw~My.
0B70:FE 20 30 F0 1C 43 4B 11   ~ 0p.CK.
0B78:40 00 21 94 11 19 10 FD   @.!....}
0B80:22 72 09 C9 CD 9E CE AF   "r.IM.N/
0B88:32 27 0E 21 1E 0F 06 10   2'.!....
0B90:CD 34 0D 21 01 0A 11 67   M4.!...g
0B98:0E CD 24 0D 21 02 0A 11   .M$.!...
0BA0:70 0E CD 24 0D 21 03 0A   p.M$.!..
0BA8:11 5F 0E CD 24 0D 0E 01   ._.M$...
0BB0:11 40 00 21 94 11 41 19   .@.!..A.
0BB8:10 FD 22 72 09 C5 CD 3B   .}"r.EM;
0BC0:0A C1 1E 11 16 00 ED 53   .A....mS
0BC8:9C 0A 3E 51 CD 95 0D CA   ..>QM..J
0BD0:DD 0A 3E 2D CD 95 0D 20   ].>-M..
0BD8:05 0D 28 D2 18 D2 ED 5B   ..(R.Rm[
0BE0:9C 0A CB 23 21 5C 0F 19   ..K#!\..
0RE8:C5 CD 9E 0A CD 5B 0A C1   EM..M[.A
0BF0:28 D2 3A 27 0E B7 20 22   (R:'.7 "
0BF8:3E 44 CD 95 0D 20 CB ED   >DM.. Km
0C00:5B 9C 0A 2A 72 09 2B 19   [..*r.+.
0C08:7E B7 28 BE 7B 32 6F 09   ~7(>{2o.
0C10:CD AE 0C 3E 01 32 27 0E   M..>.2'.
0C18:18 B0 3E 41 CD 95 0D 20   .0>AM..
0C20:A9 3A 9C 0A 32 70 09 3E   ):..2p.>
0C28:12 CD DF 0C AF 32 27 0E   .M_./2'.
0C30:CD 60 0C 30 0E C5 11 7A   M`.0.E.z
0C38:0E CD 41 0D CD 1E 0D C1   .MA.M..A
0C40:C3 CA 0B 0C CD 05 0D CD   CJ..M..M
0C48:A5 09 C5 CD 38 0A C1 3E   %.EM8.A>
0C50:21 B9 CA DD 0A C3 CA 0B   !9J].CJ.
0C58:2A 72 09 2B 19 7E B7 C9   *r.+.~7I
0C60:C5 16 00 3A 6F 09 5F CD   E..:o._M
0C68:58 0C 28 30 3A 70 09 4F   X.(0:p.O
0C70:5F CD 58 0C 20 26 3A 6F   _MX. &:o
0C78:09 5F CB 23 21 A0 0F 19   ._K#! ..
0C80:7E B9 28 1C 21 28 10 19   ~9(.!(..
0C88:7E B9 28 14 21 6C 10 19   ~9(.!l..
0C90:7E B9 28 0C 21 E4 0F 19   ~9(.!d..
0C98:7E B9 28 04 AF 3F C1 C9   ~9(./?AI
0CA0:23 7E 32 71 09 5F CD 58   #~2q._MX
0CA8:0C 28 F1 AF C1 C9 3A 6F   .(q/AI:o
```

FIGURE 26 : SUITE

```
0CB0:09 5F CB 23 21 5C 0F 19
0CB8:FD 21 B0 01 7E 3C FD 77
0CC0:02 FD 77 06 3D 3D FD 77
0CC8:04 FD 77 00 23 7E 3D FD
0CD0:77 03 FD 77 01 3C 3C FD
0CD8:77 05 FD 77 07 3E 11 21
0CE0:B0 01 C5 5F 16 04 7B 01
0CE8:00 02 CD 2F C9 15 20 F6
0CF0:C1 C9 AF CD 0A C9 28 FA
0CF8:C9 CD F7 FE CD F2 EB D7
0D00:CD 95 F5 7B C9 E5 D5 C5
0D08:FD E5 21 18 0D E5 FD 21
0D10:20 00 21 00 02 CD DF C2
0D18:FD E1 C1 D1 E1 C9 11 1C
0D20:0E 21 04 0A D5 0E 20 CD
0D28:31 C2 D1 1A B7 C8 13 CD
0D30:8A C1 18 F7 C5 3E 14 01
0D38:00 04 CD 2F C9 C1 10 F4
0D40:C9 CD 21 0D 01 00 00 0B
0D48:78 B1 20 FB C9 D9 08 DB
0D50:F1 FE 11 C2 9B C7 08 D9
0D58:ED 57 E2 C3 C3 CD BD C0
0D60:3A 2B 00 06 08 B0 32 2B
0D68:00 3E 0C D3 BB CD A2 00
0D70:C3 CF C3 E5 D5 C5 22 50
0D78:04 01 07 07 21 C4 01 CD
0D80:5F BE DD E5 E1 11 C7 01
0D88:1A FE 30 20 03 3E 20 12
0D90:CD 24 0D 18 14 E5 D5 C5
0D98:21 C8 01 77 3E 28 01 01
0DA0:01 11 C6 01 CD 2F C9 1A
0DA8:B7 C1 D1 E1 C9 00 02 77
0DB0:02 00 1D 77 1D 02 00 22
0DB8:1F 75 00 75 1F 3E 32 CD
0DC0:28 E4 3E 39 CD 28 E4 3E
0DC8:1C CD 28 E4 F3 21 99 C7
0DD0:22 3D 00 FB C3 28 C5 53
0DD8:6F 6C 69 74 61 69 72 65
0DE0:00 2D 31 20 52 65 63 68
0DE8:65 72 63 68 65 20 70 61
0DF0:72 20 53 4F 4C 2D 32 20
0DF8:56 6F 75 73 20 63 68 65
0E00:72 63 68 65 7A 0D 0A 0A
0E08:20 45 6E 74 72 65 7A 20
0E10:76 6F 74 72 65 20 63 68
0E18:6F 69 78 00 20 20 20 20
0E20:20 20 20 20 20 20 00 00
0E28:4E 69 76 65 61 75 20 64
0E30:75 20 72 65 74 6F 75 72
0E38:00 20 4F 72 64 72 65 20
```

FIGURE 26 : SUITE

```
0E40:64 20 61 6E 61 6C 79 73   d analys
0E48:65 20 20 20 20 20 28 31   e     (1
0E50:2C 32 32 29 00 76 65 72   ,22).ver
0E58:73 69 6F 6E 20 31 00 51   sion 1.Q
0E60:20 61 72 72 9B 74 00 44    arr.t.D
0E68:20 64 9C 70 61 72 74 00    d.part.
0E70:41 20 61 72 72 69 76 9C   A arriv.
0E78:65 00 49 6D 70 6F 73 73   e.Imposs
0E80:69 62 6C 65 00 20 4E 6F   ible. No
0E88:6D 62 72 65 20 64 65 20   mbre de 
0E90:70 69 6F 6E 73 20 64 65   pions de
0E98:20 6C 61 20 73 6F 6C 75    la solu
0EA0:74 69 6F 6E 00 30 20 45   tion.0 E
0EA8:66 66 61 63 65 00 31 20   fface.1 
0EB0:45 63 72 69 74 00 46 20   Ecrit.F 
0EB8:46 69 6E 00 44 27 61 63   Fin.D'ac
0EC0:63 6F 72 64 20 3F 00 4A   cord ?.J
0EC8:65 20 63 68 65 72 63 68   e cherch
0ED0:65 20 21 2E 2E 2E 00 4F   e º....O
0ED8:6E 20 72 65 63 6F 6D 6D   n recomm
0EE0:65 6E 63 65 20 28 4F 2F   ence (O/
0EE8:4E 29 00 50 61 73 20 64   N).Pas d
0EF0:65 20 73 6F 6C 75 74 69   e soluti
0EF8:6F 6E 00 20 20 4A 27 61   on.  J'a
0F00:69 20 74 72 6F 75 76 9C   i trouv.
0F08:0D 0A 0A 56 6F 69 63 69   ...Voici
0F10:20 75 6E 65 20 73 6F 6C    une sol
0F18:75 74 69 6F 6E 00 0A 02   ution...
0F20:15 02 0A 06 15 06 02 0A   ........
0F28:1D 0A 02 0E 1D 0E 02 12   ........
0F30:1D 12 02 16 1D 16 0A 1A   ........
0F38:15 1A 0A 1E 15 1E 02 0A   ........
0F40:02 15 06 0A 06 15 0A 02   ........
0F48:0A 1E 0E 02 0E 1E 12 02   ........
0F50:12 1E 16 02 16 1E 1A 0A   ........
0F58:1A 16 1E 0A 1E 16 0C 04   ........
0F60:10 04 14 04 0C 08 10 08   ........
0F68:14 08 04 0C 08 0C 0C 0C   ........
0F70:10 0C 14 0C 18 0C 1C 0C   ........
0F78:04 10 08 10 0C 10 10 10   ........
0F80:14 10 18 10 1C 10 04 14   ........
0F88:08 14 0C 14 10 14 14 14   ........
0F90:18 14 1C 14 0C 18 10 18   ........
0F98:14 18 0C 1C 10 1C 14 1C   ........
0FA0:00 00 00 1F 00 20 00 21   ...... .º
0FA8:00 01 00 02 00 03 00 15   ........
0FB0:00 16 01 04 02 05 03 06   ........
0FB8:00 1A 00 1B 00 07 00 08   ........
0FC0:04 09 05 0A 06 0B 00 0C   ........
0FC8:00 0D 07 0E 08 0F 09 10   ........
```

FIGURE 26 : SUITE

```
0FD0:0A 11 0B 12 0C 13 0D 14
0FD8:10 17 11 18 12 19 17 1C
0FE0:18 1D 19 1E 00 00 00 03
0FE8:00 01 01 02 00 06 00 04
0FF0:04 05 00 0D 00 07 07 08
0FF8:08 09 09 0A 0A 0B 0B 0C
1000:00 14 00 0E 0E 0F 0F 10
1008:10 11 11 12 12 13 00 1B
1010:00 15 15 16 16 17 17 18
1018:18 19 19 1A 00 1E 00 1C
1020:1C 1D 00 21 00 1F 1F 20
1028:00 00 09 04 0A 05 0B 06
1030:10 09 11 0A 12 0B 15 0E
1038:16 0F 17 10 18 11 19 12
1040:1A 13 1B 14 00 15 00 16
1048:1C 17 1D 18 1E 19 00 1A
1050:00 1B 00 07 00 08 1F 1C
1058:20 1D 21 1E 00 0C 00 0D
1060:00 1F 00 20 00 21 00 01
1068:00 02 00 03 00 00 03 02
1070:00 03 00 01 06 05 00 06
1078:00 04 09 08 0A 09 0B 0A
1080:0C 0B 0D 0C 00 0D 00 07
1088:10 0F 11 10 12 11 13 12
1090:14 13 00 14 00 0E 17 16
1098:18 17 19 18 1A 19 1B 1A
10A0:00 1B 00 15 1E 1D 00 1E
10A8:00 1C 21 20 00 21 00 1F
10B0:1E A0 0F E4 0F 28 10 6C
10B8:10 A0 0F E4 0F 6C 10 28
10C0:10 A0 0F 28 10 E4 0F 6C
10C8:10 A0 0F 28 10 6C 10 E4
10D0:0F A0 0F 6C 10 28 10 E4
10D8:0F A0 0F 6C 10 E4 0F 28
10E0:10 E4 0F A0 0F 28 10 6C
10E8:10 E4 0F A0 0F 6C 10 28
10F0:10 E4 0F 28 10 A0 0F 6C
10F8:10 E4 0F 28 10 6C 10 A0
1100:0F E4 0F 6C 10 28 10 A0
1108:0F E4 0F 6C 10 A0 0F 28
1110:10 28 10 A0 0F E4 0F 6C
1118:10 28 10 A0 0F 6C 10 E4
1120:0F 28 10 E4 0F A0 0F 6C
1128:10 28 10 E4 0F 6C 10 A0
1130:0F 28 10 6C 10 A0 0F E4
1138:0F 28 10 6C 10 E4 0F A0
1140:0F 6C 10 A0 0F 28 10 E4
1148:0F 6C 10 A0 0F E4 0F 28
1150:10 6C 10 E4 0F A0 0F 28
1158:10 6C 10 E4 0F 28 10 A0
```

FIGURE 26 : SUITE

```
1160:0F 01 01 01 01 01 01 01    ........
1168:01 01 01 01 01 01 01 21    ........
1170:01 00 00 00 00 00 00 00    ........
1178:00 00 00 00 00 00 00 00    ........
1180:00 00 00 00 00 00 00 00    ........
1188:00 00 00 00 00 00 00 00    ........
1190:00 00 05 03 08 01 01 03    ........
1198:03 02 0A 02 07 01 0B 04    ........
11A0:0D 02 10 04 01 03 0E 01    ........
11A8:10 04 12 02 14 02 12 04    ........
11B0:03 03 17 04 04 03 15 01    ........
11B8:17 04 19 02 1B 02 1C 04    ........
11C0:09 03 1E 04 20 04 17 01    .... ...
11C8:12 03 21 04 00 00 00 00    ..º.....
11D0:00 00 00 00 01 01 01 01    ........
```

FIGURE 26 : FIN

```
10 CLEAR50,&H7FF
20 INIT#1,"CASI:"
30 INPUT#1,N$,D,F:MOTOR
50 PRINT"Trouv : ";N$
60 FORI=D-1TOF
70 POKEI,INP(#1):NEXT:MOTOR:END
100 CLEAR50,&H7FF
110 D=&H800:F=&H11D5
120 N$="SOL":INIT#1,"CASO:"
121 INPUT"Magnto OK";T$
140 PRINT#1,N$,D,F:MOTOR
150 FORJ=1TO1800:NEXT
160 FORI=DTOF:OUT#1,PEEK(I)
170 NEXT:MOTOR:END
```

```
5 REM *** ENTREUR DE CODES ***
10 CLEAR 50,&H7FF:A=&H800
20 PRINTHEX$(A);" : ";:INPUT C$
30 V=VAL ("&H"+C$):POKE A,V
49 A=A+1:IF A>&H11D5 THEN PRINT"
...":BEEP 2,3:END
50 GOTO20
```

FIGURE 27

LES PENTOMINOS

Jouer aux **PENTOMINOS** , c'est s'attaquer à toute une série de puzzles bien particuliers . Des constructions à manier avec votre sens des mathématiques , de la logique ... et de la stratégie .

En fait , pour les non-initiés , les pentominos constituent les formes obtenues en juxtaposant le long de leurs côtés **plusieurs carrés** de surface identique . Le casse-tête utilisant les douze pentominos est assez ancien : on en trouve des énoncés dès 1907 dans des revues spécialisées telles que "CANTERBURY PUZZLES" .

Il existe donc **douze manières différentes d'assembler cinq carrés** en les joignant par leurs arêtes : les pentominos sont constitués . Pour les identifier , on utilise **leur ressemblance avec l'alphabet** .

Le problème du rangement des pentominos dans des cadres rectangulaires a été résolu . on sait qu'il n'existe que **2** solutions pour un cadre **3*20** , **368** pour un cadre **4*15** , **1010** pour un cadre **5*12** et **2339** pour un cadre **6*10** .

Malgré le grand nombre de solutions possibles et le petit nombre de pièces , la résolution de ce puzzle est difficile .

Le programme présenté a été écrit pour effectuer une résolution automatique de ce puzzle : il procède par "essais-erreurs" jusqu'à ce qu'une solution soit trouvée . Pour obtenir des solutions différentes , l'ordre d'examen des pièces est modifiable ... Notons que cet ordre conditionne la durée du calcul.

Si une imprimante X-710 est reliée au X-07 , il est possible d'imprimer les résultats obtenus .

Le mode d'emploi est très simple :

_ Possesseurs de la K7 , faites CLOAD puis RUN pour charger les codes . Sinon , rentrez les à la main avec les chargeurs fournis .

_ Le programme est lancé avec **EXEC &H800** .

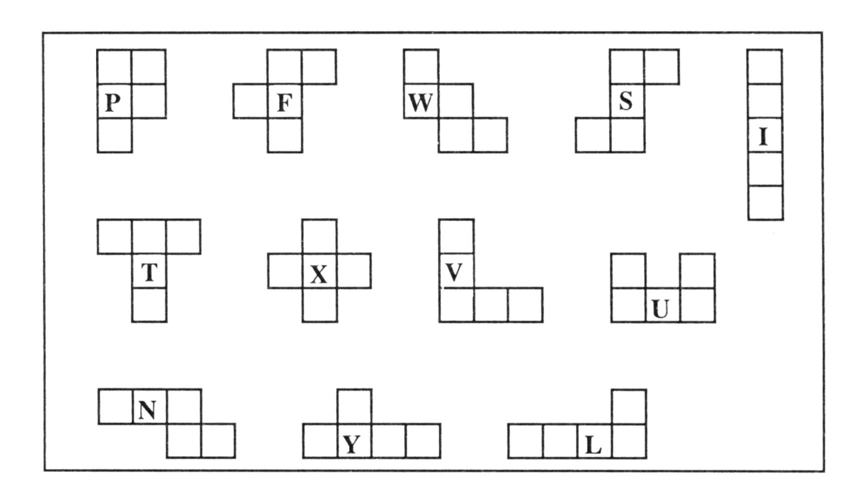
_ Tapez sur une touche après chaque affichage . Choisissez votre tableau et l'ordre d'examen de pièces .

_ Le CANON recherche la solution et vous la donne après quelques instants de réflexion . Vous pouvez alors l'imprimer sur la X-710 .

_ Pour stopper le jeu et retourner au BASIC , taper sur **"CTRL Q"** .

_ Si vous désirez une solution rapide pour les trois damiers , vous pouvez entrer la combinaison de recherche "TFILUVNWPXSY" .

Voici , pour terminer , le schéma des pièces :



Longueur de PENTOMINOS : 3489 octets .
Implantation de PENTOMINOS : de 800h à 159Ah .

```
0800:3E 33 CD 28 E4 F3 21 3E   >3M(dsº>
0808:0C 22 3D 00 FB CD BD C0   ."=.{M=@
0810:CD 9E CE CD 08 0C 06 04   M.NM....
0818:21 7B 10 CD 31 0C 21 02   º{.M1.º.
0820:06 11 61 0E CD 21 0C CD   ..a.Mº.M
0828:01 0C CD 9E CE CD 08 0C   ..M.NM..
0830:21 61 0F CD F7 FE CD 01   ºa.Mw~M.
0838:0C CD 9E CE 21 AB 0F CD   .M.Nº+.M
0840:F7 FE AF 21 C0 11 11 C1   w~/º@..A
0848:11 77 01 2F 00 ED B0 32   .w./.m02
0850:BF 11 3C 32 74 11 06 05   ?.<2t...
0858:AF 21 00 0E 11 01 0E E5   /º.....e
0860:D5 C5 77 01 60 00 ED B0   UEw.`.m0
0868:C1 D1 E1 24 14 10 F0 3E   AQa$..p>
0870:20 21 8E 10 11 8F 10 01    º......
0878:66 00 77 ED B0 CD 01 0C   f.wm0M..
0880:CD 9E CE 21 AF 0D CD F7   M.Nº/.Mw
0888:FE CD 01 0C D6 30 11 61   ~M..V0.a
0890:12 FE 01 28 10 11 6A 12   .~.(..j.
0898:FE 02 28 09 11 73 12 FE   ~.(..s.~
08A0:03 28 02 18 E4 1A 32 DB   .(..d.2[
08A8:08 32 77 0B 13 13 1A 32   .2w....2
08B0:DE 08 32 83 0B 13 1A 32   ^.2....2
08B8:E0 08 32 85 0B 13 1A 32   `.2....2
08C0:B4 0B 13 1A 32 6E 0B 13   4...2n..
08C8:1A 32 6F 0B 13 1A 32 7A   .2o...2z
08D0:09 13 1A 32 7B 09 CD 9E   ...2{.M.
08D8:CE AF 21 96 10 0E 0C 06   N/º.....
08E0:05 77 23 10 FC 23 23 0D   .w#.¦##.
08E8:20 F5 CD 08 0C 21 79 0E    uM..ºy.
08F0:CD F7 FE 21 03 01 11 6C   Mw~º...l
08F8:0E CD 21 0C 21 04 01 22   .Mº.º.."
0900:61 10 11 A1 0E CD 21 0C   a..º.Mº.
0908:21 6C 0E 11 63 10 01 0C   ºl..c...
0910:00 ED B0 06 0C CD 01 0C   .m0..M..
0918:0E 0C 21 62 10 23 BE 28   ..ºb.#>(
0920:05 0D 20 F9 18 EF 36 00   .. y.o6.
0928:2A 61 10 C5 4F CD 31 C2   *a.EOM1B
0930:24 22 61 10 C1 3E 0C 91   $"a.A>..
0938:16 00 5F 21 6F 10 19 3E   .._ºo..>
0940:0C 90 77 2E 03 63 24 0E   ..w..c$.
0948:20 C5 CD 31 C2 C1 10 C5    EM1BA.E
0950:CD 01 0C CD 9E CE 21 AE   M..M.Nº.
0958:0E CD F7 FE 16 00 DD 21   .Mw~..]º
0960:81 11 FD 21 6F 10 06 0C   ..}ºo...
0968:21 75 11 FD 5E 00 19 DD   ºu.}^..]
0970:7E 00 77 DD 23 FD 23 10   ~.w]#}#.
0978:EF 11 7C 12 FD 21 6F 10   o.¦.}ºo.
0980:DD 21 81 11 AF 32 72 11   ]º../2r.
0988:DD 7E 00 DD 23 3C 32 71   ]~.]#<2q
```

FIGURE 28 : PENTOMINOS

```
0990:11 AF 32 73 11 01 00 00
0998:C5 CD 61 11 06 05 26 0E
09A0:1A 77 13 24 10 FA C1 04
09A8:3A 73 11 3C 32 73 11 3A
09B0:71 11 B8 20 E3 FD 23 3A
09B8:72 11 3C 32 72 11 FE 0C
09C0:20 C6 16 00 3A BF 11 3C
09C8:32 BF 11 21 8E 10 5F AF
09D0:19 BE 20 EE 32 BE 11 3A
09D8:BE 11 5F AF 21 CC 11 19
09E0:BE CA 4D 0A 3A BE 11 3C
09E8:32 BE 11 3A BE 11 FE 0C
09F0:38 E5 3A 74 11 3D 32 74
09F8:11 5F 21 D8 11 19 7E 32
0A00:BE 11 21 E4 11 19 7E 32
0A08:73 11 21 C0 11 19 7E 32
0A10:BD 11 3A BE 11 5F AF 21
0A18:CC 11 19 77 3A BE 11 CD
0A20:64 11 5D D5 26 0E 7E 47
0A28:3A BD 11 4F 80 5F 32 BF
0A30:11 21 8E 10 19 36 00 D1
0A38:CD 07 0B 06 04 21 8E 10
0A40:DD 5E 00 19 36 00 DD 23
0A48:10 F3 C3 81 0A AF 32 73
0A50:11 3A BE 11 CD 64 11 5D
0A58:26 0E 46 3A BF 11 90 32
0A60:BD 11 4F CD 07 0B AF 06
0A68:03 21 8E 10 DD 5E 00 19
0A70:BE 20 0E DD 23 10 F2 21
0A78:8E 10 DD 5E 00 19 BE 28
0A80:21 3A 73 11 3C 32 73 11
0A88:3A BE 11 5F 21 75 11 19
0A90:3A 73 11 BE 28 BB 38 B9
0A98:3A BE 11 3C 32 BE 11 C3
0AA0:EB 09 3A 74 11 FE 0C 20
0AA8:03 C3 26 0B 3A BF 11 5F
0AB0:21 8E 10 19 3A 74 11 77
0AB8:3A BE 11 CD 64 11 5D 3A
0AC0:BD 11 4F CD 07 0B 3A 74
0AC8:11 06 04 21 8E 10 DD 5E
0AD0:00 19 77 DD 23 10 F4 21
0AD8:CC 11 3A BE 11 5F 19 36
0AE0:01 3A 74 11 5F 21 D8 11
0AE8:19 3A BE 11 77 21 E4 11
0AF0:19 3A 73 11 77 21 C0 11
0AF8:19 3A BD 11 77 3A 74 11
0B00:3C 32 74 11 C3 C2 09 DD
0B08:21 F0 11 26 0F 6B 7E 81
0B10:DD 77 00 24 7E 81 DD 77
0B18:01 24 7E 81 DD 77 02 24
```

FIGURE 28 : SUITE

```
0B20:7E 81 CD 77 03 C9 CD 08   ~.Mw.IM.
0B28:0C CD 9E CE 21 D4 0E CD   .M.N!T.M
0B30:F7 FE CD 01 0C CD 6A 0B   w~M..Mj.
0B38:21 02 0F 11 8D 15 CD 21   !.....M!
0B40:0C 21 03 0F 11 94 15 CD   .!.....M
0B48:21 0C CD 01 0C FE 4F CC   !.M..~OL
0B50:00 15 CD 9E CE 21 C0 0E   ..M.N!@.
0B58:CD F7 FE CD 01 0C FE 4E   Mw~M..~N
0B60:CA 64 0C C3 42 08 30 43   Jd.CB.0C
0B68:45 39 CD 9E CE 21 9D 11   E9M.N!..
0B70:06 04 CD 31 0C FD 21 96   ..M1.}!.
0B78:10 3E FB 32 8B 10 AF 32   .>{2../2
0B80:8C 10 16 0C 1E 05 3E 1E   ......>.
0B88:32 8D 10 06 05 3A 8C 10   2....:..
0B90:80 32 8C 10 3A 8B 10 80   .2..:...
0B98:32 8B 10 06 FB 3A 8D 10   2...{:..
0BA0:80 32 8D 10 FD 7E 00 FD   .2..}~.}
0BA8:BE 01 28 03 CD DF 0B FD   >.(.M_.}
0BB0:7E 00 FD BE 07 28 03 CD   ~.}>.(.M
0BB8:C7 0B FD 23 1D 20 DC FD   G.}#. \}
0BC0:23 FD 23 15 20 BE C9 21   #}#. >I!
0BC8:C4 01 E5 3A 8C 10 77 23   D.e:..w#
0BD0:23 77 2B 3A 8D 10 77 06   #w+:..w.
0BD8:05 80 23 23 77 18 16 21   ..##w..!
0BE0:C4 01 E5 3A 8B 10 77 23   D.e:..w#
0BE8:23 06 05 80 77 2B 3A 8D   #...w+:.
0BF0:10 77 23 23 77 E1 D5 01   .w##waU.
0BF8:00 04 3E 14 CD 2F C9 D1   ..>.M/IQ
0C00:C9 AF CD 0A C9 28 FA C9   I/M.I(zI
0C08:E5 D5 C5 FD E5 21 1B 0C   eUE}e!..
0C10:E5 FD 21 20 00 21 00 02   e}! .!..
0C18:CD DF C2 FD E1 C1 D1 E1   M_B}aAQa
0C20:C9 D5 0E 20 CD 31 C2 D1   IU. M1BQ
0C28:1A B7 C8 13 CD 8A C1 18   .7H.M.A.
0C30:F7 C5 3E 14 01 00 04 CD   wE>....M
0C38:2F C9 C1 10 F4 C9 D9 08   /IA.tIY.
0C40:DB F1 FE 11 C2 9B C7 08   [q~.B.G.
0C48:D9 ED 57 E2 C3 C3 CD BD   YmWbCCM=
0C50:C0 3A 2B 00 06 08 B0 32   @:+...02
0C58:2B 00 3E 0C D3 BB CD A2   +.>.S;M"
0C60:00 C3 CF C3 3E 32 CD 28   .COC>2M(
0C68:E4 F3 21 99 C7 22 3D 00   ds!.G"=.
0C70:FB C3 28 C5 00 01 02 09   {C(E....
0C78:0A 00 01 02 08 09 01 02   ........
0C80:08 09 0A 00 01 09 08 0A   ........
0C88:01 08 09 10 11 00 08 09   ........
0C90:10 11 00 01 08 09 11 00   ........
0C98:01 08 09 10 00 08 09 0A   ........
0CA0:11 02 08 09 0A 11 01 08   ........
0CA8:09 0A 10 01 08 09 0A 12   ........
```

FIGURE 28 : SUITE

```
0CB0:01 02 08 09 11 00 01 09
0CB8:0A 11 01 08 09 11 12 01
0CC0:09 0A 10 11 00 01 09 0A
0CC8:0B 02 03 08 09 0A 01 02
0CD0:03 08 09 00 01 02 0A 0B
0CD8:02 08 09 11 19 01 08 09
0CE0:10 18 01 09 10 11 18 00
0CE8:08 10 11 19 00 08 09 0A
0CF0:0B 03 08 09 0A 0B 00 01
0CF8:02 03 08 00 01 02 03 0B
0D00:00 01 09 11 19 00 01 08
0D08:10 18 01 09 11 18 19 00
0D10:08 10 18 19 01 08 09 0A
0D18:0B 02 08 09 0A 0B 00 01
0D20:02 03 09 00 01 02 03 0A
0D28:00 08 09 10 18 01 08 09
0D30:11 19 00 08 10 11 18 01
0D38:09 10 11 19 00 08 09 11
0D40:12 02 09 0A 10 11 01 02
0D48:08 09 10 00 01 09 0A 12
0D50:00 08 09 0A 12 02 08 09
0D58:0A 10 01 02 09 10 11 00
0D60:01 09 11 12 00 08 10 11
0D68:12 02 0A 10 11 12 00 01
0D70:02 08 10 00 01 02 0A 12
0D78:01 09 10 11 12 00 01 02
0D80:09 11 00 08 09 0A 10 02
0D88:08 09 0A 12 00 02 08 09
0D90:0A 00 01 02 08 0A 00 01
0D98:08 10 11 00 01 09 10 11
0DA0:00 01 02 03 04 00 08 10
0DA8:18 20 01 08 09 0A 11 43
0DB0:68 6F 69 73 69 73 73 65
0DB8:7A 20 6C 65 20 44 61 6D
0DC0:69 65 72 20 31 20 2D 3E
0DC8:20 36 2A 31 30 0D 0A 20
0DD0:32 20 2D 3E 20 35 2A 31
0DD8:32 0D 0A 20 33 20 2D 3E
0DE0:20 34 2A 31 35 00 09 41
0DE8:46 46 49 43 48 62 02 01
0DF0:3B 6C 02 0B 44 49 4E 49
0DF8:54 09 58 4F 52 09 41 76
0E00:01 00 00 00 00 00 01 00
0E08:00 01 00 00 00 00 00 02
0E10:00 02 00 03 02 02 02 02
0E18:00 01 00 01 00 01 00 00
0E20:00 00 01 01 00 00 00 01
0E28:00 00 02 00 00 00 00 00
0E30:00 01 00 01 00 00 00 00
0E38:00 00 00 00 00 00 00 00
```

FIGURE 28 : SUITE

```
0E40:00 01 00 00 00 01 00 00   ........
0E48:00 00 00 01 00 01 00 01   ........
0E50:00 00 00 00 00 00 00 01   ........
0E58:00 00 00 00 00 01 00 00   ........
0E60:00 50 65 6E 74 61 6D 69   .Pentami
0E68:6E 6F 73 00 50 46 4E 4C   nos.PFNL
0E70:59 57 53 56 54 55 49 58   YWSVTUIX
0E78:00 20 43 68 6F 69 73 69   . Choisi
0E80:73 73 65 7A 20 6C 27 6F   ssez l'o
0E88:72 64 72 65 20 64 27 65   rdre d'e
0E90:78 61 6D 65 6E 20 64 65   xamen de
0E98:73 20 70 69 9A 63 65 73   s pi,ces
0EA0:00 2E 2E 2E 2E 2E 2E 2E   ........
0EA8:2E 2E 2E 2E 2E 00 0A 20   .......
0EB0:4A 65 20 63 68 65 72 63   Je cherc
0EB8:68 65 20 21 2E 2E 2E 00   he !....
0EC0:4F 6E 20 72 65 63 6F 6D   On recom
0EC8:6D 65 6E 63 65 20 28 4F   mence (O
0ED0:2F 4E 29 00 20 20 4A 27   /N).  J'
0ED8:61 69 20 74 72 6F 75 76   ai trouv
0EE0:9C 0D 0A 0A 56 6F 69 63   ....Voic
0EE8:69 20 75 6E 65 20 73 6F   i une so
0EF0:6C 75 74 69 6F 6E 00 06   lution..
0EF8:09 49 4E 43 09 48 5C 03   .INC.H\.
0F00:07 01 01 07 02 01 08 07   ........
0F08:07 07 07 07 00 01 07 07   ........
0F10:01 03 01 07 09 07 07 08   ........
0F18:02 02 00 07 01 02 07 01   ........
0F20:01 01 02 02 00 01 00 07   ........
0F28:01 01 07 01 01 01 01 01   ........
0F30:00 07 00 02 00 07 00 01   ........
0F38:00 07 00 00 00 07 00 01   ........
0F40:00 07 00 01 00 07 00 00   ........
0F48:00 01 00 07 00 08 00 08   ........
0F50:00 07 00 00 00 01 00 08   ........
0F58:00 07 00 07 00 08 00 00   ........
0F60:00 20 49 6C 20 65 78 69   . Il exi
0F68:73 74 65 20 31 32 20 70   ste 12 p
0F70:65 6E 74 61 2D 6D 69 6E   enta-min
0F78:6F 73 20 65 74 20 32 33   os et 23
0F80:33 39 20 66 61 E5 6F 6E   39 faeon
0F88:73 64 65 20 6C 65 73 20   sde les
0F90:72 61 6E 67 65 72 20 64   ranger d
0F98:61 6E 73 20 20 75 6E 20   ans  un
0FA0:63 61 64 72 65 20 36 2A   cadre 6*
0FA8:31 30 00 20 31 30 31 30   10. 1010
0FB0:20 66 61 E5 6F 6E 73 20    faeons
0FB8:64 61 6E 73 20 75 6E 20   dans un
0FC0:63 61 64 72 65 20 35 2A   cadre 5*
0FC8:31 32 2C 20 65 74 20 33   12, et 3
```

FIGURE 28 : SUITE

```
0FD0:36 38 20 64 61 6E 73 20   68 dan
0FD8:75 6E 20 63 61 64 72 65   un cad
0FE0:20 34 2A 31 35 00 44 54    4*15.
0FE8:1A 04 0C 09 4C 44 09 44   ....LD
0FF0:45 2C 44 54 35 31 32 24   E,DT51
0FF8:04 05 09 43 50 09 32 2E   ...CP.
1000:08 02 02 08 07 08 0E 08   ......
1008:0E 08 08 08 00 02 0E 08   ......
1010:02 07 02 08 0E 08 08 09   ......
1018:00 08 00 08 07 03 08 02   ......
1020:02 02 08 07 00 09 00 08   ......
1028:08 02 08 02 02 02 08 08   ......
1030:00 08 00 07 00 08 00 08   ......
1038:00 08 00 00 00 08 00 08   ......
1040:00 08 00 07 00 08 00 00   ......
1048:00 07 00 08 00 0E 00 0F   ......
1050:00 0E 00 00 00 07 00 09   ......
1058:00 0E 00 0E 00 0E 00 00   ......
1060:00 04 00 00 00 00 00 00   ......
1068:00 00 02 00 00 00 00 01   ......
1070:03 05 07 09 0B 0A 08 06   ......
1078:04 02 00 00 02 77 02 00   .....w
1080:1D 77 1D 02 00 02 1F 75   .w....
1088:00 75 1F 37 3C 05 20 20   .u.7<.
1090:20 20 20 20 20 20 01 01
1098:01 02 02 20 20 03 01 01   ...
10A0:04 02 20 20 03 03 04 04   ..
10A8:02 20 20 03 04 04 05 02   .
10B0:00 20 03 06 05 05 05 20
10B8:00 06 06 06 05 07 20 20   ....
10C0:05 08 07 07 07 20 20 06   ...
10C8:08 09 09 07 20 20 08 0A
10D0:0A 09 0B 20 20 08 0A 08   ...
10D8:09 0B 20 20 0A 0A 0B 0B
10E0:0B 20 20 00 00 00 00 00   .
10E8:20 20 20 20 20 20 20 20
10F0:20 20 20 20 20 04 0A 09
10F8:4C 44 09 41 2C 28 44 45   LD,A,(
1100:09 08 03 09 03 09 0F 09   ......
1108:0F 09 09 0F 00 07 15 09   ......
1110:07 08 08 09 0F 09 09 0E   ......
1118:00 09 00 09 0E 07 09 03   ......
1120:07 03 0E 08 00 08 00 09   ......
1128:0E 09 09 03 09 03 0F 09   ......
1130:00 0E 00 08 00 0F 00 0F   ......
1138:00 0E 00 00 00 0E 00 09   ......
1140:00 0E 00 0E 00 0F 00 00   ......
1148:00 08 00 0F 00 0F 00 13   ......
1150:00 0F 00 00 00 08 00 0E   ......
1158:00 0F 00 15 00 0F 00 00   ......
```

FIGURE 28 : SUITE

```
160:00 FD 7E 00 47 3A 73 11   .}~.G:s.
168:87 87 4F 87 87 91 80 6F   ..O....o
170:C9 01 0C 00 0C 00 07 01   I.......
178:07 03 07 03 07 03 07 03   ........
180:03 07 07 07 07 07 03 03   ........
188:03 03 03 01 00 00 00 32   .......2
190:00 00 1E 32 1E 00 01 00   ...2....
198:1D 32 01 32 1D 00 05 3C   .2.2...<
1A0:05 00 1E 3C 1E 00 06 00   ...<....
1A8:1D 3C 06 3C 1D 00 0A 4B   .<.<...K
1B0:0A 00 1E 4B 1E 00 0B 00   ...K....
1B8:1D 4B 0B 4B 1D 55 02 55   .K.K.U.U
1C0:00 08 0B 0F 10 1F 24 2D   ......$-
1C8:32 3B 40 42 01 01 00 01   2;@B....
1D0:01 01 01 01 01 01 01 01   ........
1D8:00 01 07 09 0B 00 03 06   ........
1E0:05 04 0A 08 00 00 04 04   ........
1E8:01 00 02 23 25 03 02 01   ........
1F0:56 57 58 59 45 52 2B 39   VWXYER+9
1F8:AA 05 0B 49 44 32 09 4C   *..ID2.L
200:0F 09 04 0F 09 0A 10 0A   ........
208:10 0A 10 10 00 08 1C 0F   ........
210:09 09 0F 0A 10 0A 0E 0F   ........
218:00 07 00 0E 0F 08 0E 07   ........
220:0E 08 0F 0E 00 07 00 10   ........
228:0F 0A 10 0A 10 09 10 10   ........
230:00 0F 00 0F 00 16 00 16   ........
238:00 15 00 00 00 0F 00 0F   ........
240:00 15 00 15 00 16 00 00   ........
248:00 0F 00 10 00 15 00 16   ........
250:00 15 00 00 00 0E 00 0F   ........
258:00 16 00 16 00 16 00 00   ........
260:00 97 10 0A 06 08 8D 11   ........
268:74 0C 96 10 0C 05 07 9D   t.......
270:11 7C 12 95 10 0F 04 06   .|......
278:AD 11 B7 13 00 01 02 08   -.7.....
280:09 00 01 02 07 08 01 02   ........
288:08 09 07 00 01 09 06 07   ........
290:01 07 08 0E 0F 00 07 08   ........
298:0E 0F 00 01 07 08 0F 00   ........
2A0:01 07 08 0E 00 07 03 09   ........
2A8:0F 02 07 08 09 0F 01 07   ........
2B0:08 09 0E 01 07 08 09 10   ........
2B8:01 02 07 08 0F 00 01 08   ........
2C0:09 0F 01 07 08 0F 10 01   ........
2C8:08 09 0E 0F 00 01 08 09   ........
2D0:0A 02 03 07 08 09 01 02   ........
2D8:03 07 08 00 01 02 09 0A   ........
2E0:00 07 08 0F 10 01 07 08   ........
2E8:0E 15 01 08 0E 0F 15 00   ........
```

FIGURE 28 : SUITE

```
12F0:07 0E 0F 16 00 07 08 09
12F8:0A 03 07 08 09 0A 00 01
1300:02 03 07 00 01 02 03 0A
1308:00 01 08 0F 16 00 01 07
1310:0E 15 01 08 0F 15 16 00
1318:07 0E 15 16 01 07 08 09
1320:0A 02 07 08 09 0A 00 01
1328:02 03 08 00 01 02 03 09
1330:00 07 08 0E 15 01 07 08
1338:0F 16 00 07 0E 0F 15 01
1340:08 0E 0F 16 00 07 08 0F
1348:10 02 08 09 0E 0F 01 02
1350:07 08 0E 00 01 08 09 10
1358:00 07 08 09 10 02 07 08
1360:09 0E 01 02 08 0E 0F 00
1368:01 08 0F 10 00 07 0E 0F
1370:10 02 09 0E 0F 10 00 01
1378:02 07 0E 00 01 02 09 10
1380:01 08 0E 0F 10 00 01 02
1388:08 0F 00 07 08 09 0E 02
1390:07 08 09 10 00 02 07 08
1398:09 00 01 02 07 09 00 01
13A0:07 0E 0F 00 01 08 0E 0F
13A8:00 01 02 03 04 00 07 0E
13B0:15 1C 01 07 08 09 0F 00
13B8:01 02 07 08 00 01 02 06
13C0:07 01 02 06 07 08 00 01
13C8:06 07 08 01 06 07 0C 0D
13D0:00 06 07 0C 0D 00 01 06
13D8:07 0D 00 01 06 07 0C 00
13E0:06 07 08 0D 02 06 07 08
13E8:0D 01 06 07 08 0C 01 06
13F0:07 08 0E 01 02 06 07 0D
13F8:00 01 07 08 0D 01 06 07
1400:0D 0E 01 07 08 0C 0D 00
1408:01 07 08 09 02 03 06 07
1410:08 01 07 03 06 07 00 01
1418:02 08 09 00 06 07 0D 13
1420:01 06 07 0C 12 01 07 0C
1428:0D 12 00 06 0C 0D 13 00
1430:06 07 08 09 03 06 07 08
1438:09 00 01 02 03 06 00 01
1440:02 03 09 00 01 07 0D 13
1448:00 01 06 0C 12 01 07 0D
1450:12 13 00 06 0C 12 13 01
1458:06 07 08 09 02 06 07 08
1460:09 00 01 02 03 07 00 01
1468:02 03 08 00 06 07 0C 12
1470:01 06 07 0D 13 00 06 0C
1478:0D 12 01 07 0C 0D 13 00
```

FIGURE 28 : SUITE

```
1480:06 07 0D 0E 02 07 08 0C   ........
1488:0D 01 02 06 07 0C 00 01   ........
1490:07 08 0E 00 06 07 08 0E   ........
1498:02 06 07 08 0C 01 02 07   ........
14A0:0C 0D 00 01 07 0D 0E 00   ........
14A8:06 0C 0D 0E 02 08 0C 0D   ........
14B0:0E 00 01 02 06 0C 00 01   ........
14B8:02 08 0E 01 07 0C 0D 0E   ........
14C0:00 01 02 07 0D 00 06 07   ........
14C8:08 0C 02 06 07 08 0E 00   ........
14D0:02 06 07 08 00 01 02 06   ........
14D8:08 00 01 06 0C 0D 00 01   ........
14E0:07 0C 0D 00 01 02 03 04   ........
14E8:00 06 0C 12 18 01 06 07   ........
14F0:08 0D 43 41 4C 4C 09 43   ..CALL.C
14F8:4C 53 20 08 18 3B 2A 2A   LS ..;**
1500:21 76 15 CD 60 15 3A E0   !v.M`.:`
1508:08 FE 06 01 00 33 28 0A   .~...3(.
1510:FE 05 01 05 3D 28 03 01   ~...=(..
1518:0A 4C 78 32 48 15 79 32   .Lx2H.y2
1520:23 15 0E 05 06 00 21 6E   #.....!n
1528:02 70 23 71 2B C5 3E 10   .p#q+E>.
1530:01 01 02 11 72 02 CD 2F   ....r.M/
1538:C9 1A B7 3E 4F 28 02 3E   I.7>O(.>
1540:20 CD F7 CE C1 04 78 FE    MwNA.x~
1548:3D 20 DB C5 21 6B 15 CD   = [E!k.M
1550:60 15 C1 0C 79 FE 1F 20   `.A.y~.
1558:CB 21 80 15 CD 60 15 C9   K!..M`.I
1560:7E B7 C8 E5 CD F7 CE E1   ~7HeMwNa
1568:23 18 F5 0D 0A 12 4D 30   #.u...M0
1570:2C 36 0D 41 0D 00 12 53   ,6.A...S
1578:30 0D 43 31 0D 41 0D 00   0.C1.A..
1580:0D 0A 12 4D 30 2C 2D 36   ...M0,-6
1588:30 0D 41 0D 00 53 6F 72   0.A..Sor
1590:74 69 65 00 58 37 31 30   tie.X710
1598:20 3F 00 00 00 00 00 00    ?......
```

FIGURE 28 : FIN

```
10 CLEAR50,&H7FF
20 INIT#1,"CASI:"
30 INPUT#1,N$,D,F:MOTOR
50 PRINT"Trouv : ";N$
60 FORI=D-1TOF
70 POKEI,INP(#1):NEXT:MOTOR:END
100 CLEAR50,&H7FF
110 D=&H800:F=&H159A
120 N$="PENTA":INIT#1,"CASO:"
121 INPUT"Magnto OK";T$
140 PRINT#1,N$,D,F:MOTOR
150 FORJ=1TO1800:NEXT
160 FORI=DTOF:OUT#1,PEEK(I)
170 NEXT:MOTOR:END


5 REM *** ENTREUR DE CODES ***
10 CLEAR50,&H7FF:A=&H800
20 PRINTHEX$(A);" : ";:INPUT C$
30 V=VAL("&H"+C$):POKE A,V
49 A=A+1:IF A>&H159A THEN PRINT"
...":BEEP2,3:END
50 GOTO20
```

FIGURE 29 : CHARGEURS

AUTONUM

Ce programme implémente la fonction AUTO ... Ce qui signifie que le CANON va disposer **d'une numérotation automatique des lignes de programme BASIC** !!

Après avoir chargé le programme "AUTONUM" dans la mémoire du X-07 , il suffit de choisir **l'adresse de fin de la routine** à implanter suivant la configuration de votre CANON . Ensuite , un FSET judicieux protègera votre routine (FSET 215 minimum) .

La procédure d'implantation de la routine résidente va pouvoir être débutée . Pour cela , faire RUN suivi de RETURN ... Le logiciel vous demande votre adresse de fin (en hexadécimal) et l'adresse de lancement vous est renvoyée . Le chargeur devenu inutile peut être détruit par l'ordre NEW .

L'utilisation d'AUTONUM est la suivante :

_ En tapant **CTRL N** , le programme affiche un point d'interrogation . Vous devez alors entrer le numéro de la première ligne et l'incrément choisi . Exemple : en tapant , "100,10" suivi de RETURN , la séquence générée sera 100, 110 , 120 , 130 ...

_ L'appui sur **CTRL Z** agit comme une bascule permettant d'entrer et de ressortir de la fonction AUTO .

_ **CTRL Q** désactive de manière définitive AUTONUM . Pour le relancer , il faut alors taper EXEC AD , AD représentant l'adresse de lancement affichée au tout début des opérations .

AUTONUM fonctionne grâce **à la modification de l'interruption clavier** . Si une pression est détectée sur la touche RETURN , le logiciel agit de la manière suivante :

_ Exécution de l'opération demandée .

_ Si AUTONUM est actif , écrire dans le tampon d'entrée le numéro de la ligne et l'incrément (pour la ligne suivante) .

_ Revenir à l'exécution normale .

Voici quelques remarques sur **la programmation** de AUTONUM ...

Tout d'abord , ce logiciel utilise cinq octets de mémoire morte pour stocker ses données ... Ces octets sont situés dans **le tampon utilisé par les routines CSAVE et CLOAD** .

Ensuite , AUTONUM agit par interception du vecteur des interruptions (&H3D) . Or le programme est relogeable : il faut donc trouver une astuce pour que ce logiciel effectue de lui-même la modification correcte de ce vecteur . Une solution est décrite ci-dessous :

```
Origine : quelconque

Début   LD HL , $D5D1        ; Ecriture en $300 de la séquence :
        LD ($300) , (HL)     ;             POP DE
        LD A , $C9           ;             PUSH DE
        LD ($302) , A        ;             RET
        CALL  $300           ; Lors du CALL , l'adresse XXX est empilée

XXX     LD HL , 14           ; Au retour du CALL , DE = XXX
        ADD HL , DE          ; HL = AUTONUM (14 octets entre XXX et
                               AUTONUM)
        DI
        LD ($3D) , HL        ; Nouveau vecteur d'interruption
        EI
        XOR A                ; Inactiver la routine AUTONUM : l'adresse
        LD ($300) , A        ; $300 contient le drapeau activé par la
        RET                  ; séquence CTRL Z

AUTO.   Début du logiciel ...
```

<u>Longueur de AUTONUM (BASIC)</u> : 1093 octets .
<u>Longueur de la routine</u> : 215 octets .
<u>Implantation de la routine</u> : relogeable .

```
1 REM fonction AUTONUM pour X-07
10 CLEAR50,&HFFFF:DEFINTA-Z:CLS
20 INPUT"Adr. fin ";A$:F=VAL("&H"+A$)
30 D=F-215:PRINT"Adr. deb = ";HEX$(D)
40 FORI=DTOF:READA:POKEI,A:NEXT
50 EXECD:END
10000 DATA033,209,213,034,000,003,062,20
1,050,002,003,205,000,003,033,014
10010 DATA000,025,243,034,061,000,251,17
5,050,000,003,201,217,008,219,242
10020 DATA230,001,202,193,200,219,240,23
0,192,040,008,230,128,202,053,200
10030 DATA195,012,200,219,241,254,013,03
2,075,095,175,205,098,194,175,205
10040 DATA170,194,058,000,003,183,202,18
9,200,062,001,211,245,217,008,197
10050 DATA213,229,245,237,091,001,003,04
2,003,003,025,034,001,003,235,205
10060 DATA156,187,035,205,110,213,205,00
4,215,205,059,202,010,197,213,095
10070 DATA175,205,098,194,175,205,170,19
4,209,193,021,003,032,238,241,225
10080 DATA209,193,251,201,254,026,032,02
7,062,001,211,245,217,008,197,213
10090 DATA229,245,062,013,239,062,010,23
9,033,000,003,175,182,047,119,032
10100 DATA221,024,176,254,014,032,041,06
2,001,211,245,217,008,197,213,229
10110 DATA245,175,050,000,003,205,242,23
5,215,205,204,255,237,083,001,003
10120 DATA207,044,205,204,255,237,083,00
3,003,175,061,050,000,003,024,131
10130 DATA254,017,194,128,200,195,195,19
5
```

FIGURE 30 : AUTONUM

REFBAS

REFBAS est un utilitaire qui permet d'obtenir soit une liste triée de toutes les variables d'un programme BASIC , soit une table croisée des références à un numéro de ligne . Ce logiciel est donc particulièrement utile si l'on désire analyser , modifier les noms des variables , réorganiser ou documenter un programme BASIC , surtout si ce dernier est long .

Le mode d'emploi de REFBAS est le suivant :

_ Avant de charger REFBAS , faire **FSET 1024** . Cette opération permettra la protection ultérieure de REFBAS situé en fin de mémoire fichier (version **16 Ko**) . Carte moniteur en place : **FSET 4096** .

_ Charger REFBAS en tapant CLOAD suivi de RETURN puis RUN toujours suivi de RETURN . Les codes se chargeront et le programme sera lancé par la commande **EXEC &H3C10** . Comme toujours , si vous ne possédez pas la K7 , les chargeurs sont à votre disposition .

_ Un menu s'affiche alors avec trois options disponibles :

* OPTION V : REFBAS construit une table triée de toutes les variables et des numéros de lignes où ces variables sont appelées .
* OPTION S : le programme édifie une table des références croisées pour les instructions GOTO , GOSUB , THEN , ELSE , RESUME , RUN et RESTORE . Le numéro des lignes ciblées est imprimé en tête ainsi que le numéro des lignes faisant référence à cette cible .
* OPTION B : retour au BASIC .

_ Remarquons que la commande **CTRL Q** permet de revenir au BASIC à n'importe quel moment et que **BREAK** est actif durant la phase d'impression .

Pendant son fonctionnement , REFBAS crée à la suite du programme à analyser une table utilisant **six octets par variable ou référence** . Quand cette table arrive au contact de la pile , l'utilitaire annonce que la taille mémoire est insuffisante : il convient dans ce cas d'analyser le programme étudié en le segmentant .

Pour terminer , voici une petite note technique pour les fans de la programmation avancée ...

La longueur des lignes (40 pour la X-710) est contenue à l'adresse **&H3EDB** . Si vous désirez imprimer en plus petit , libre à vous !!

Tous les BASIC issus de la firme MICROSOFT sont structurés de la même manière :

_ 2 octets de chaînage indiquant l'adresse absolue de la ligne suivante .

_ 2 octets pour le numéro de ligne .

_ n octets de code . Les mots clés du BASIC sont codés sur un ou deux octets (un seul pour le X-07) afin de limiter l'encombrement mémoire et de faciliter la tâche de l'interpréteur . Les autres caractères sont codés en ASCII .

_ 1 octet égal à 0 représentant l'indicateur de fin de ligne .

L'adresse de la zone programme est contenue dans l'octet **&HB2** (indicateur TXTTAB) et est égal , en principe , à &H553 . Vous pouvez utilement consulter les "Mystères du X-07" pour plus de précisions .

La boucle de scrutation du programme BASIC de REFBAS (similaire à celle d'EXABAS et LLIST) est structurée de la manière suivante :

```
         LD IX , (TXTTAB)     ;  Initialisation
BOUCLE   XOR A                ; A = 0
         CP (IX + 1)          ; Si (IX + 1) = 0 , c'est la fin du programme
         JP Z , fin           ; Fin
         PUSH IX              ; Sauve l'adresse du début de la ligne
         LD L , (IX + 0)      ; L'adresse de la ligne suivante est stockée
         LD H , (IX + 1)      ; dans le registre HL
         LD E , (IX + 2)      ; Le numéro de ligne est stocké dans le
         LD D , (IX + 3)      ; registre DE
         PUSH HL
         POP IX               ; IX pointe sur la ligne suivante
         POP HL               ; On récupère l'adresse du début de la ligne
         LD BC , 4
         ADD HL , BC          ; HL contient le premier code de la ligne
         .........................  ;  Traitement
         JP BOUCLE
```

REFBAS se livre à una analyse détaillée des différents types de variables et envisage tous les problèmes de chaîne . En ce qui concerne les références , il va les chercher derrière les mots clés .

<u>Longueur de REFBAS</u> : 985 octets .

<u>Implantation de REFBAS</u> : de 3C10 h à 3FE9h .

```
3C10:2A DD 01 F9 F3 21 B1 3D  *].ysº1=
3C18:22 3D 00 FB CD 9E CE 21  "=.{M.Nº
3C20:2A 3F CD F7 FE AF CD 0A  *?Mw~/M.
3C28:C9 28 FA FE 56 28 13 FE  I(z~V(.~
3C30:53 28 28 FE 42 20 EE F3  S((~B ns
3C38:21 99 C7 22 3D 00 FB C3  º.G"=.{C
3C40:28 C5 21 EA 3C 22 B7 3C  (Eºj<"7<
3C48:3E 06 32 15 3D 21 30 3D  >.2.=º0=
3C50:22 2E 3D 21 AE 3F 22 1D  ".=º.?".
3C58:3E 18 17 21 66 3D 22 B7  >..ºf="7
3C60:3C 3E 66 32 15 3D 21 9A  <>f2.=º.
3C68:3D 22 2E 3D 21 C4 3F 22  =".=ºD?"
3C70:1D 3E CD 9E CE 21 67 3F  .>M.Nºg?
3C78:CD F7 FE AF CD 0A C9 28  Mw~/M.I(
3C80:FA CD 9E CE CD B7 CF 21  zM.NM7Oº
3C88:75 3F CD F7 FE ED 5B 22  u?Mw~m["
3C90:03 CD 96 3E DD 2A B2 00  .M.>]*2.
3C98:AF DD BE 01 28 6B DD E5  /]>.(k]e
3CA0:E1 01 04 00 09 7E FE 00  a....~~.
3CA8:28 1C FE 83 28 38 FE 8E  (.~.(8~.
3CB0:28 14 FE 22 28 03 C3 66  (.~"(.Cf
3CB8:3D 23 7E FE 00 28 07 FE  =#~~.(.~
3CC0:22 20 F6 23 18 DF DD 6E  " v#._]n
3CC8:00 DD 66 01 E5 DD E1 18  .]f.e]a.
3CD0:C7 23 23 7E FE 00 28 EE  G##~~.(n
3CD8:FE 30 38 E7 FE 3A 38 F2  ~08g~:8r
3CE0:FE 41 38 DF FE 47 38 EA  ~A8_~G8j
3CE8:18 D9 FE 26 28 E3 FE 41  .Y~&(c~A
3CF0:38 D1 FE 5B 30 CD E5 21  8Q~[0Meº
3CF8:28 00 19 ED 72 E1 DA 0F  (..mraZ.
3D00:3D D5 21 88 3F CD F7 FE  =Uº.?Mw~
3D08:D1 CD 96 3E C3 D7 3D D5  QM.>CW=U
3D10:FD E1 FD 36 00 66 DD 7E  }a}6.f]~
3D18:02 FD 77 01 DD 7E 03 FD  .}w.]~.}
3D20:77 02 FD 75 04 FD 74 05  w.}u.}t.
3D28:0E 00 0C 23 7E C3 9A 3D  ...#~C.=
3D30:FE 21 28 F6 FE 25 28 F2  ~º(v~%(r
3D38:FE 23 28 EE FE 24 28 EA  ~#(n~$(j
3D40:FE 28 20 04 0C 23 18 10  ~( ..#..
3D48:FE 30 38 0C FE 3A 38 DA  ~08.~:8Z
3D50:FE 41 38 04 FE 5B 38 D2  ~A8.~[8R
3D58:FD 71 03 7B C6 06 5F 7A  }q.{F._z
3D60:CE 00 57 C3 A5 3C FE 88  N.WC%<~.
3D68:28 19 FE 89 28 15 FE 8C  (.~.(.~.
3D70:28 11 FE 90 28 0D FE 99  (.~.(.~.
3D78:28 09 FE 8B 28 05 FE CE  (.~.(.~N
3D80:C2 C3 3C 23 7E FE 20 28  BC<#~~ (
3D88:FA FE 2C 28 F6 FE 30 DA  z~,(v~0Z
3D90:A5 3C FE 3A D2 A5 3C C3  %<~:R%<C
3D98:F6 3C FE 30 38 05 FE 3A  v<~08.~:
```

FIGURE 31 : REFBAS

```
3DA0:DA 2A 3D FD 71 03 7B C6
3DA8:06 5F 7A CE 00 57 C3 84
3DB0:3D D9 08 DB F1 FE 11 C2
3DB8:9B C7 08 D9 ED 57 E2 C3
3DC0:C3 CD BD C0 3A 2B 00 06
3DC8:08 B0 32 2B 00 3E 0C D3
3DD0:BB CD A2 00 C3 C3 C3 21
3DD8:9F 3F CD F7 FE 2A 22 03
3DE0:E5 DD E1 01 06 00 09 E5
3DE8:FD E1 AF FD BE 00 28 2C
3DF0:CD A3 3E FA F8 3D 20 05
3DF8:FD E5 E1 18 E3 06 06 DD
3E00:56 00 FD 5E 00 DD 73 00
3E08:FD 72 00 DD 23 FD 23 10
3E10:EE DD E5 E1 01 0C 00 B7
3E18:ED 42 18 C4 21 C4 3F CD
3E20:D4 3E CD 9E CE 2A 22 03
3E28:11 06 00 19 22 DB 3F CD
3E30:B0 CF DD 2A DB 3F AF 32
3E38:0D 3F DD BE 00 28 50 DD
3E40:22 DD 3F 06 00 DD 4E 03
3E48:11 E0 3F DD 6E 04 DD 66
3E50:05 ED B0 21 DF 3F CD D4
3E58:3E 01 08 00 11 E1 3F 21
3E60:E0 3F 36 20 ED B0 FD 2A
3E68:DB 3F FD 6E 01 FD 66 02
3E70:CD 0E 3F CD D4 3E 2A DB
3E78:3F 11 06 00 19 22 DB 3F
3E80:DD 2A DD 3F FD 2A DB 3F
3E88:CD A3 3E 28 D9 18 A0 CD
3E90:B0 CF B7 C3 1C 3C D5 E1
3E98:13 01 05 00 36 00 ED B0
3EA0:36 B0 C9 DD 56 05 DD 5E
3EA8:04 FD 66 05 FD 6E 04 DD
3EB0:7E 03 FD 4E 03 DD CB 00
3EB8:76 28 02 B9 C0 B9 30 01
3EC0:4F 1A ED A1 C0 AF B1 28
3EC8:03 13 18 F5 DD 7E 03 FD
3ED0:4E 03 B9 C9 46 04 3A 0D
3ED8:3F 80 FE 28 32 0D 3F 38
3EE0:19 E5 CD B0 CF 06 07 0E
3EE8:20 C5 79 CD F7 CE C1 10
3EF0:F8 E1 46 04 3E 07 80 32
3EF8:0D 3F 23 05 C5 E5 7E CD
3F00:F7 CE E1 C1 23 10 F5 3E
3F08:20 CD F7 CE C9 00 22 50
3F10:04 21 A1 01 01 07 07 CD
3F18:5F BE 3E 30 21 A0 01 06
3F20:05 20 23 BE 20 02 10 F9
3F28:2B C9 52 45 46 42 41 53
```

FIGURE 31 : SUITE

```
F30:20 76 2E 31 36 4B 0D 0A  v.16K..
F38:56 20 3D 20 4C 69 73 74 V = List
F40:65 20 56 61 72 69 61 62 e Variab
F48:6C 65 73 20 53 20 3D 20 les S = 
F50:4C 69 73 74 65 20 53 61 Liste Sa
F58:75 74 73 0D 0A 42 20 3D uts..B =
F60:20 42 61 73 69 63 00 58  Basic.X
F68:2D 37 31 30 20 70 72 65 -710 pre
F70:74 65 20 3F 00 41 6E 61 te ?.Ana
F78:6C 79 73 65 20 65 6E 20 lyse en 
F80:63 6F 75 72 73 0D 0A 00 cours...
F88:4D 65 6D 6F 69 72 65 20 Memoire 
F90:69 6E 73 75 66 66 69 73 insuffis
F98:61 6E 74 65 0D 0A 00 54 ante...T
FA0:72 69 20 65 6E 20 63 6F ri en co
FA8:75 72 73 0D 0A 00 15 4C urs....L
FB0:69 73 74 65 20 64 65 73 iste des
FB8:20 76 61 72 69 61 62 6C  variabl
FC0:65 73 0D 0A 16 54 61 62 es...Tab
FC8:6C 65 20 64 65 73 20 72 le des r
FD0:65 66 65 72 65 6E 63 65 eference
FD8:73 0D 0A 61 06 D3 06 06 s..a.S..
FE0:20 20 20 20 20 20 20 20
FE8:20 00 00 00 00 00 00 00  .......
```

FIGURE 31 : FIN

```
5 REM *** CHARGEUR REFBAS 16 Ko ***
10 CLEAR50,&H9FF
20 INIT#1,"CASI:"
30 INPUT#1,N$,D,F
40 MOTOR
50 PRINT"Trouv :";N$
60 FORI=D-1TOF
70 POKEI,INP(#1)
80 NEXT:MOTOR
90 END
100 CLEAR50,&H9FF
110 D=&H3C10:F=&H3FE9
120 N$="REFBAS":INIT#1,"CASO:"
130 INPUT"Magnto OK";T$
140 PRINT#1,N$,D,F:MOTOR
150 FORI=1TO1800:NEXT
160 FORI=DTOF:OUT#1,PEEK(I)
170 NEXT:MOTOR
180 END
```

```
5 REM *** ENTREUR DE CODES ***
10 CLEAR50,&H9FF:A=&H3C10
20 PRINTHEX$(A);" : ";:INPUT C$
30 V=VAL("&H"+C$):POKE A,V
49 A=A+1:IF A>&H3FE9 THEN PRINT"TER
...":BEEP2,3:END
50 GOTO20
```

```
ACTION DE REFBAS SUR LE CHARGEUR


Liste des variables

D      30 60 110 140 160
F      30 60 110 140 160
I      60 70 150 160 160
N$     30 50 120 140
T$     130
```

FIGURE 32: EXEMPLE+CHARGEURS

EXABAS

Dans la continuité de REFBAS , voici EXABAS qui constitue aussi un utilitaire puissant en implémentant les fonctions **REF** (références) et **SR** (Search and Replace) que l'on trouve sur certains gros ordinateurs dotés de fonctions d'éditeur infiniment plus complètes et plus performantes ...

EXABAS constitue donc un utilitaire permettant de trouver dans un programme BASIC **toutes les occurences d'une variable , d'une chaîne , d'un mot-clé . Il permet également de remplacer une chaîne quelconque ou un mot-clé par une autre chaîne ou un autre mot-clé.**

Le mode d'emploi d'EXABAS est aussi simple que celui de REFBAS :

_ Avant de charger EXABAS , faire **FSET 1024** (ou **FSET 4096** si une carte moniteur est en place) pour le protéger .
_ Charger EXABAS de la même manière que REFBAS .
_ Lancer EXABAS par la commande **EXEC &H3C10** .
_ Un menu comportant six options s'affiche alors :
* OPTION B : sortie du programme et retour au BASIC .
* OPTION V : recherche d'une variable particulière .
* OPTION C : recherche d'une chaîne quelconque (sauf les mots-clés ...) .
* OPTION T : recherche d'un mot-clé quelconque .

Pour ces précédentes options , EXABAS réclame la chaîne à identifier puis affiche les numéros de ligne où cette chaîne figure .
* OPTION c : remplacement d'une chaîne par une autre .
* OPTION t : remplacement d'un mot-clé par un autre .

Pour ces deux dernières options , EXABAS réclame le nom de la chaîne à remplacer puis celle de remplacement . Il affiche ensuite les numéros des lignes où l'opération est effectuée .
_ L'appui sur **CTRL Q** permet de revenir instantanément au BASIC .
_ Dans tous les cas , le logiciel marque une pose quand l'afficheur est rempli . Il attend la pression sur une touche pour continuer l'exécution . Si cette touche est **"I"** , le contenu de l'écran sera envoyé vers la X-710 . Notons qu'une fois l'analyse terminée , le X-07 affiche un carreau .

EXABAS permet quantité d'applications . C'est un outil puissant qui a été testé sans problème sur de nombreux programmes . Néanmoins , il est indispensable d'effectuer une copie de protection des programmes à étudier avant d'utiliser EXABAS .

Au niveau technique , les problèmes de recherche sont traités de la même manière que REFBAS . Le problème du remplacement des chaînes est plus délicat à traiter car la longueur du programme est susceptible de varier . Aussi, après chaque modification , est il nécessaire d'ajuster tous les pointeurs des lignes suivant celle qui vient d'être modifiée .

Longueur d' EXABAS : 901 octets .
Implantation d' EXABAS : de 3C10h à 3F95h .

```
3C10:2A DD 01 F9 F3 21 50 3C   *].ys!P<
3C18:22 3D 00 FB CD 9E CE 21   "=.{M.N!
3C20:ED 3C CD F7 FE CD 76 3C   m<Mw~Mv<
3C28:FE 56 CA 1D 3E FE 54 CA   ~VJ.>~TJ
3C30:85 3D FE 43 CA 95 3D FE   .=~CJ.=^
3C38:74 CA 3F 3F FE 63 CA 62   tJ??~cJb
3C40:3E FE 42 20 E0 F3 21 99   >~B `s!.
3C48:C7 22 3D 00 FB C3 28 C5   G"=.{C(E
3C50:D9 08 DB F1 FE 11 C2 9B   Y.[q~.B.
3C58:C7 08 D9 ED 57 E2 C3 C3   G.YmWbCC
3C60:CD BD C0 3A 2B 00 06 08   M=@:+...
3C68:B0 32 2B 00 3E 0C D3 BB   02+.>.S;
3C70:CD A2 00 C3 C3 C3 AF CD   M".CCC/M
3C78:0A C9 28 FA C9 CD F7 FE   .I(zIMw^
3C80:CD F2 EB 23 11 D8 3C 0E   Mrk#.X<.
3C88:00 7E 12 B7 28 08 23 13   .~.7(.#.
3C90:0C 79 FE 14 38 F3 79 FE   .y~.8sy^
3C98:00 28 E2 32 EC 3C C9 22   .(b2l<I"
3CA0:50 04 21 A1 01 01 07 07   P.!!....
3CA8:CD 5F BE 3E 30 21 A0 01   M_>>0! .
3CB0:06 05 70 23 BE C0 10 FA   ..p#>@.z
3CB8:C9 E5 21 63 02 06 50 3E   Ie!c..P>
3CC0:20 BE 20 03 2B 10 FA 04    > .+.z.
3CC8:21 14 02 E5 C5 7E CD F7   !..eE~Mw
3CD0:CE C1 E1 23 10 F5 E1 C9   NAa#.uaI
3CD8:54 24 00 6E 65 74 6F 00   T$.neto.
3CE0:20 20 20 20 20 20 20 20
3CE8:20 20 20 20 02 45 78 61       .Exa
3CF0:62 61 73 20 76 2E 31 36   bas v.16
3CF8:4B 0D 0A 56 3D 76 61 72   K..V=var
3D00:2E 20 20 20 20 54 3D 6D   .    T=m
3D08:6F 74 20 63 6C 9C 20 43   ot cl. C
3D10:3D 63 68 61 69 6E 65 20   =chaine
3D18:20 42 3D 42 61 73 69 63    B=Basic
3D20:0D 0A 63 2F 74 3D 43 68   ..c/t=Ch
3D28:61 6E 67 65 6D 65 6E 74   angement
3D30:00 4D 6F 74 20 63 6C 9C   .Mot cl.
3D38:00 43 68 61 69 6E 65 00   .Chaine.
3D40:4E 6F 6D 00 49 6E 69 74   Nom.Init
3D48:69 61 6C 0D 0A 00 4E 6F   ial...No
3D50:75 76 65 61 75 0D 0A 00   uveau...
3D58:20 20 20 20 20 20 20 20
3D60:20 20 20 20 20 20 20 20
3D68:20 20 20 20 00 CD 9E CE       .M.N
3D70:21 31 3D CD F7 FE CD F2   !1=Mw~Mr
3D78:EB 23 CD 2A F3 3A D3 00   k#M*s:S.
3D80:FE 7F 38 E9 C9 CD 6D 3D   ~.8iIMm=
3D88:32 D8 3C AF 32 D9 3C 3C   2X</2Y<<
3D90:32 EC 3C 18 09 CD 9E CE   2l<..M.N
3D98:21 39 3D CD 7C 3C DD 2A   !9=M|<]*
```

FIGURE 33 : EXABAS

```
3DA0:B2 00 CD 81 3F 7E FE 00   2.M.?
3DA8:28 12 1A BE CC C1 3D 23   (..>L
3DB0:18 F3 CD 6E 00 DD 66 01   .s]n.
3DB8:E5 DD E1 C9 CD B2 3D 18   e]aIM
3DC0:E1 E5 D5 3A EC 3C 47 1A   aeU:l
3DC8:BE 20 06 23 13 10 F8 18   > .#.
3DD0:03 D1 E1 C9 C1 C1 E5 DD   .QaIA
3DD8:66 03 DD 6E 02 CD 9F 3C   f.]n.
3DE0:3A B8 00 FE 04 20 0E 2B   :8.~.
3DE8:46 23 3A B9 00 80 FE 12   F#:9.
3DF0:38 03 CD 02 3E CD F7 FE   8.M.>
3DF8:3E 20 CD BE C1 E1 11 D8   > M>A
3E00:3C C9 CD 76 3C FE 49 CC   <IMv<
3E08:B9 3C CD 9E CE C9 E1 3E   9<M.N
3E10:83 CD BE C1 CD 02 3E CD   .M>AM
3E18:B0 CF C3 1C 3C CD 9E CE   0OC.<
3E20:21 40 3D CD 7D 3C DD 2A   !@=M}
3E28:B2 00 CD 81 3F 7E FE 00   2.M.?
3E30:28 1B FE 83 28 17 FE 8E   (.~.(
3E38:28 13 FE 22 28 02 18 12   (.~"(
3E40:23 7E FE 00 28 07 FE 22   #~~.(
3E48:20 F6 23 18 E0 CD B2 3D    v#.`
3E50:18 D8 FE 41 38 F4 FE 5B   .X~A8
3E58:30 F0 1A BE CC C1 3D 23   0p.>L
3E60:18 CB CD 9E CE 21 44 3D   .KM.N
3E68:CD 7D 3C 21 4E 3D CD F7   M}<!N
3E70:FE CD F2 EB 23 11 58 3D   ~Mrk#
3E78:0E 00 7E 12 B7 28 08 23   ..~.7
3E80:13 0C 79 FE 14 38 F3 79   ..y~.
3E88:FE 00 28 DF 32 6C 3D CD   ~.(_2
3E90:9E CE 3A EC 3C 4F 3A 6C   .N:l<
3E98:3D 81 32 93 3F 3E 00 DE   =.2.?
3EA0:00 32 94 3F DD 2A B2 00   .2.?]
3EA8:CD 81 3F 7E FE 00 20 03   M.?~~
3EB0:CD B2 3D 18 F3 1A BE 28   M2=.s
3EB8:03 23 18 EF 11 D8 3C E5   .#.o.
3EC0:3A EC 3C 47 1A BE 20 06   :l<G.
3EC8:23 13 10 F8 18 07 E1 23   #..x.
3ED0:11 D8 3C 18 D6 3A 93 3F   .X<.V
3ED8:E1 B7 20 12 EB 06 00 3A   a7 .k
3EE0:6C 3D 4F 21 58 3D ED B0   l=O!X
3EE8:EB CD D6 3D 18 BD CB 7F   kMV=.
3EF0:28 29 E5 E5 FD E1 06 00   ()ee}
3EF8:3A EC 3C 4F 09 E5 3A 6C   :l<O.
3F00:3D 4F FD 09 FD E5 D1 C1   =O}.}
3F08:E5 2A 22 03 AF ED 42 44   e*"./
3F10:4D E1 ED B0 CD 5D 3F E1   Mam0M
3F18:C3 DC 3E E5 06 00 3A EC   C\>e.
3F20:3C 4F 09 EB 2A 22 03 AF   <O.k*
3F28:ED 52 44 4D 2A 93 3F ED   mRDM*
```

FIGURE 33 : SUITE

```
3F30:5B 22 03 1B 19 EB ED B8  [",..km
3F38:CD 5D 3F E1 C3 DC 3E CD  M]?aC\>
3F40:6D 3D 32 D8 3C 21 4E 3D  m=2X<!N
3F48:CD F7 FE CD 70 3D 32 58  Mw~Mp=2
3F50:3D 3E 01 32 EC 3C 32 6C  =>.2l<2
3F58:3D AF C3 9A 3E DD E5 ED  =/C.>]e
3F60:4B 93 3F AF DD 6E 00 DD  K.?/]n.
3F68:66 01 09 DD 75 00 DD 74  f..]u.]
3F70:01 E5 DD E1 DD BE 01 20  .e]a]>.
3F78:EB DD E1 23 23 22 22 03  k]a##""
3F80:C9 AF 11 D8 3C DD BE 01  I/.X<]>
3F88:CA 0E 3E DD E5 E1 01 04  J.>]ea.
3F90:00 09 C9 00 00 66 00 00  ..I..f.
```

FIGURE 33 : FIN

```
10 CLEAR50,&H9FF
20 INIT#1,"CASI:"
30 INPUT#1,N$,D,F
40 MOTOR
50 PRINT"Trouv :";N$
60 FORI=D-1TOF
70 POKEI,INP(#1)
80 NEXT:MOTOR
90 END
100 CLEAR50,&H9FF
110 D=&H3C10:F=&H3F95
120 N$="EXABAS":INIT#1,"CASO:"
130 INPUT"Magnto OK";T$
140 PRINT#1,N$,D,F:MOTOR
150 FORI=1TO1800:NEXT
160 FORI=DTOF:OUT#1,PEEK(I)
170 NEXT:MOTOR
180 END
```

```
5 REM *** ENTREUR DE CODES ***
10 CLEAR50,&H9FF:A=&H3C10
20 PRINTHEX$(A);" : ";:INPUT C$
30 V=VAL("&H"+C$):POKE A,V
40 A=A+1:IF A>&H3F95 THEN PRINT"
...":BEEP2,3:END
50 GOTO20
```

```
ACTION DE EXABAS SUR LE CHARGEUR

Nom? N                 30 50 120 140
Mot cl? PRINT          50 140

Mot cl? FOR            60 150 160
Mot cl? DEFSTR
Nom? T$                130
```

FIGURE 34: CHARGEURS

LE PIEGE

Enfin **un vrai jeu d'aventures** pour le CANON X-07 !!! Ce genre de jeu était vraiment très rare pour ne pas dire inexistant et nous avons décidé de vous y faire goûter ... Vous n'allez pas être déçu !

Le but du jeu est d'explorer une grande maison : la visite de toutes les salles est indispensable pour atteindre la sortie ... Bonne chance !!

Le mode d'emploi est le suivant :

_ Charger le programme en **version vidéo ou LCD** suivant vos désirs . Comme toujours , le petit programme BASIC se chargera de rentrer les codes en mémoire . Faire un FSET avant le chargement tout en sachant que la mémoire est altérée de **&H800 à &H1900** .

_ le programme possède trois points d'entrée :

* **EXEC &H800** : début du jeu .
* **EXEC &H815** : vous repartez de l'endroit quitté par la commande "P" .
* **EXEC &H838** : vous repartez de l'endroit quitté par la commande "Q" .

_ Le programme affiche sur la ligne n° ...

* 1 : le nom de la salle et les directions possibles (N , S , E , W) .
* 2 : les objets mobiles éventuellement présents dans cette salle .
* 3 : un message de description de la salle .
* 4 : ligne utilisée pour recevoir vos commandes et afficher les réponses du programme .

_ Les commandes se répartissent en deux catégories :

* composées de une lettre :
  _ N , S , E , W pour se déplacer .
  _ I pour avoir la liste des objets possédés .
  _ Q pour quitter .
  _ P pour faire une pause (sauvegarde) .
* normales : le programme comprend des "phrases" composées d'un verbe à l'infinitif suivi d'un nom . Ex : "SAUTER COULOIR" . En fait , seules les quatre premières lettres de chaque mot sont analysées et il suffit de taper dans l'exemple "SAUT COUL" . Si le verbe n'est pas compris , "????" est affiché . Par contre , si le nom n'est pas compris , "   ????" apparait.

<u>Longueur des logiciels</u> : 4205 octets (LCD) et 4329 octets (vidéo) .
<u>Implantation</u> : de 800h à 186Ch (LCD) / de 800h à 18E8h (vidéo) .

```
0800:CD 9E CE DD 21 F3 17 CD   M.N]♀s.M
0808:BB 08 21 52 08 11 75 08   ;.♀R..u.
0810:01 23 00 ED B0 21 75 08   .#.m0♀u.
0818:11 16 0C 01 1F 00 ED B0   ......m0
0820:3A 94 08 32 47 15 3A 95   :..2G.:.
0828:08 32 86 15 3A 97 08 32   .2..:..2
0830:EF 0E 3A 96 08 32 D2 0F   o.:..2R.
0838:F3 21 2A 09 22 3D 00 FB   s♀*."=.
0840:CD 9E CE 21 88 17 CD F7   M.N♀..Mw
0848:FE CD 21 09 CD 70 09 C3   ~M♀.Mp.C
0850:0D 0A 01 00 00 00 00 00   .......
0858:00 00 00 00 00 00 00 00   .......
0860:00 00 0F 04 0A 17 02 08   .......
0868:08 0E 0F 05 00 02 16 00   .......
0870:FF 06 00 14 57 01 00 00   ....W..
0878:00 00 00 00 00 00 00 00   .......
0880:00 00 00 00 00 0F 04 0A   .......
0888:17 02 08 08 0E 0F 05 00   .......
0890:02 16 00 FF 06 00 14 57   .......
0898:02 01 02 0B 03 01 03 0B   .......
08A0:04 01 04 0B CD BD C0 AF   ....M=@
08A8:CD 0A C9 28 FA C9 DD 21   M.I(zI]
08B0:B6 08 CD BB 08 C9 17 02   6.M;.I.
08B8:20 02 FF DD 7E 00 FE FF    ..]~.~
08C0:C8 5F DD 46 01 16 00 4A   H_]F...
08C8:DD 23 DD 23 21 D8 08 E5   ]#]#♀X.
08D0:C5 D5 FD E1 E1 CD DF C2   EU}aaM_
08D8:2A 0E 00 7C B5 20 F9 18   *..|5 y
08E0:DA 11 FA 08 21 04 01 22   Z.z.♀..
08E8:B8 00 D5 0E 20 CD 31 C2   8.U. M1
08F0:D1 1A B7 C8 13 CD 8A C1   Q.7H.M.
08F8:18 F7 20 20 20 20 20 20   .w
0900:20 20 20 20 20 20 20 20
0908:20 20 20 20 20 00 C5 3E        .E
0910:14 01 00 04 CD 2F C9 C1   ....M/I
0918:C9 46 23 CD 0E 09 10 FB   IF#M...
0920:C9 01 00 00 0B 78 B1 20   I....x1
0928:FB C9 D9 08 DB F1 FE 11   {IY.[q~
0930:C2 9B C7 08 D9 ED 57 E2   B.G.YmW
0938:C3 C3 CD BD C0 3A 2B 00   CCM=@:+
0940:06 08 B0 32 2B 00 3E 0C   ..02+.>
0948:D3 BB CD A2 00 C3 CF C3   S;M".CO
0950:21 61 09 18 03 21 63 09   ♀a...♀c
0958:3E 07 01 00 02 CD 2F C9   >....M/
0960:C9 03 03 00 03 F3 21 99   I....s♀
0968:C7 22 3D 00 FB C3 28 C5   G"=.{C(
0970:CD 9E CE CD AE 08 CD BD   M.NM..M
0978:C0 21 FD 09 CD 19 09 21   @♀}.M..
0980:04 08 22 B8 00 21 06 0A   .."8.♀.
0988:CD F7 FE 0E 0C 2E 02 CD   Mw~....
```

FIGURE 35 : LE PIEGE (LCD)

```
0990:0A C9 C0 C5 01 00 35 CD
0998:24 09 C1 79 FE 0B 38 0D
09A0:CD D8 09 0C 79 FE 15 20
09A8:E6 0E 01 18 E2 C5 3E 15
09B0:91 67 41 11 F1 09 1A 4F
09B8:D5 C5 CD 31 C2 C1 D1 13
09C0:24 10 F3 26 01 1A FE 00
09C8:28 0A 4F D5 CD 31 C2 D1
09D0:13 24 18 F1 C1 0C 18 B7
09D8:C5 3E 15 91 67 11 F1 09
09E0:1A 4F FE 00 28 09 D5 CD
09E8:31 C2 D1 13 24 18 F1 C1
09F0:C9 2A 4C 45 20 50 49 45
09F8:47 45 2A 20 00 02 00 05
0A00:77 05 00 11 77 11 84 85
0A08:84 85 84 85 00 CD 4F 0C
0A10:CD E5 0C CD 80 0D 3A 16
0A18:0C FE 1A CA E4 17 FE 09
0A20:28 F9 FE 19 CA CE 17 3A
0A28:23 0C FE 01 20 57 3A 6E
0A30:02 FE 49 CA AB 0D FE 51
0A38:CA E9 0D FE 50 CA F7 0D
0A40:0E 00 FE 4E 28 17 0C FE
0A48:53 28 12 0C FE 45 28 0D
0A50:0C FE 57 28 08 11 06 0B
0A58:CD CF 0A 18 B6 3A 16 0C
0A60:47 11 04 00 21 28 15 19
0A68:05 20 FC 59 19 AF BE 20
0A70:08 11 14 0B CD CF 0A 18
0A78:9A 7E 32 16 0C 11 2C 0B
0A80:CD CF 0A 18 88 CD E1 0A
0A88:FE 00 20 05 11 06 0B 18
0A90:E3 32 30 0B 1A FE 00 28
0A98:F3 FE 20 28 03 13 18 F4
0AA0:13 EB 11 6E 02 01 05 00
0AA8:ED B0 21 DC 11 CD E4 0A
0AB0:FE 00 20 05 11 0B 0B 18
0AB8:BB 32 31 0B 3A 30 0B 87
0AC0:26 00 6F 11 67 12 19 5E
0AC8:23 56 EB 3A 31 0B E9 D5
0AD0:CD E1 08 D1 CD E4 08 CD
0AD8:AE 08 CD 21 09 CD E1 08
0AE0:C9 21 2D 0B 01 05 00 09
0AE8:AF BE C8 E5 DD E1 11 6E
0AF0:02 01 04 00 1A ED A1 20
0AF8:08 13 EA F4 0A 7E EE 80
0B00:C9 DD E5 E1 18 DE 3F 3F
0B08:3F 3F 00 20 20 20 20 3F
0B10:3F 3F 3F 00 49 6D 70 6F
0B18:73 73 69 62 6C 65 00 31
```

FIGURE 35 : SUITE

```
0B20:34 32 30 39 37 32 31 38   4209721
0B28:31 30 38 39 4F 2E 4B 00   10890.K
0B30:01 0F 4F 55 56 52 81 50   ..OUVR.
0B38:52 45 4E 82 50 4F 53 45   REN.POS
0B40:83 4C 41 43 48 83 44 45   .LACH.D
0B48:53 43 84 42 4F 49 52 85   SC.BOIR
0B50:4C 49 52 45 8E 4F 54 45   LIRE.OT
0B58:52 87 45 4E 4C 45 87 53   R.ENLE.
0B60:41 55 54 88 52 45 47 41   AUT.REG
0B68:89 45 58 41 4D 89 54 49   .EXAM.T
0B70:52 45 8A 50 4F 55 53 8A   RE.POUS
0B78:41 4C 4C 55 8B 46 52 41   ALLU.FR
0B80:50 8C 43 41 53 53 8C 46   P.CASS.
0B88:45 52 4D 8D 4D 41 4E 47   ERM.MAN
0B90:8E 4D 4F 4E 54 8F 41 4C   .MONT.A
0B98:4C 45 90 45 4E 54 52 91   LE.ENTR
0BA0:45 43 4F 55 92 45 54 45   ECOU.ET
0BA8:49 93 4D 45 54 54 94 46   I.METT.
0BB0:41 49 52 95 00 00 43 4C   AIR...C
0BB8:45 46 00 54 4F 52 43 48   EF.TORC
0BC0:45 00 50 41 50 49 45 52   E.PAPIE
0BC8:00 43 41 53 53 45 54 54   .CASSET
0BD0:45 00 43 4F 52 44 45 00   E.CORDE
0BD8:43 41 46 45 00 4D 41 52   CAFE.MA
0BE0:54 45 41 55 00 43 4C 45   TEAU.CL
0BE8:53 00 42 55 52 49 4E 00   S.BURIN
0BF0:4C 49 56 52 45 00 54 45   LIVRE.T
0BF8:4E 41 49 4C 4C 45 53 00   NAILLES
0C00:50 49 4C 45 53 00 50 4F   PILES.P
0C08:4D 4D 45 53 00 53 45 52   MMES.SE
0C10:52 55 52 45 00 FF 08 00   RURE...
0C18:00 00 00 00 00 00 00 00   .......
0C20:00 00 00 01 02 00 0F 04   .......
0C28:0A 17 02 08 08 0E 0F 05   .......
0C30:00 02 16 00 FF 23 7E FE   .....#~
0C38:00 20 FA 05 20 F7 23 EB   . z. w#
0C40:C9 3A 24 0C FE 02 C8 3C   I:$.~.H
0C48:32 24 0C CD D1 0D C9 3A   2$.MQ.I
0C50:25 0C FE 00 28 11 3A 18   %.~.(.:
0C58:0C 3C 32 18 0C FE 20 20   .<2..~
0C60:06 3E 1A 32 16 0C C9 3A   .>.2..I
0C68:1A 0C FE 00 28 0B 3A 1B   ..~.(.:
0C70:0C FE 02 28 04 3E 09 18   .~.(.>.
0C78:EA 3A 1B 0C FE 02 20 22   j:..~.
0C80:3A 17 0C 3C 32 17 0C FE   :..<2..
0C88:0F 20 17 CD 9E CE 21 D5   . .M.N!
0C90:0C CD F7 FE 21 27 0C AF   .Mw~!'.
0C98:77 32 1B 0C CD 21 09 C3   w2..M!.
0CA0:21 09 3A 16 0C FE 06 20   !.:..~.
0CA8:13 3A 22 0C FE 00 C0 32   .:".~.@
```

FIGURE 35 : SUITE

```
0CB0:47 15 3C 32 22 0C 3E 20   G.<2".>
0CB8:32 EF 0E C9 FE 12 20 04   2o.I~.
0CC0:32 25 0C C9 FE 0E C0 3A   2%.I~.@
0CC8:20 0C 3C 32 20 0C FE 03    .<2 .~
0CD0:C0 E1 C3 E4 17 0A 50 69   @aCd..P
0CD8:6C 65 73 20 9C 70 75 69   les .pu
0CE0:73 9C 65 73 00 CD 9E CE   s.es.M.
0CE8:3A 16 0C 47 21 7B 0E CD   :..G!{.
0CF0:35 0C 21 00 00 CD EA 08   5.!..Mj
0CF8:AF FD 21 98 08 32 24 0C   /}!..2$
0D00:47 3A 16 0C 4F 21 26 0C   G:..O!&
0D08:7E FE FF 28 08 04 23 B9   ~~.(..#
0D10:CC 41 0C 18 F3 3A 16 0C   LA..s:.
0D18:47 AF 32 1A 0C 78 CD 2E   G/2..xM
0D20:0D 21 0B 10 CD 35 0C 21   .!..M5.
0D28:03 01 CD EA 08 C9 FE 05   ..Mj.I~
0D30:20 09 3A 19 0C FE 00 C8    .:..~.
0D38:06 1D C9 FE 0D 28 12 FE   ..I~.(.
0D40:10 28 0E FE 11 28 0A FE   .(.~.(.
0D48:14 28 06 FE 15 28 02 18   .(.~.(.
0D50:0E 3E 01 32 1A 0C 3A 1B   .>.2..:
0D58:0C FE 02 C8 06 1B C9 FE   .~.H..I
0D60:0F 20 09 3A 1C 0C FE 01   . .:..~
0D68:C0 06 1C C9 FE 07 C0 3A   @..I~.@
0D70:22 0C FE 00 C8 06 06 3A   ".~.H..
0D78:21 0C FE 06 C0 06 1E C9   !.~.@..
0D80:CD E1 08 CD BD C0 CD 50   Ma.M=@M
0D88:09 CD F2 EB 23 11 6E 02   .Mrk#.n
0D90:0E 00 7E 12 B7 28 08 23   ..~.7(.
0D98:13 0C 79 FE 14 38 F3 79   ..y~.8s
0DA0:FE 00 28 DC 32 23 0C CD   ~.(\2#.
0DA8:55 09 C9 CD 9E CE FD 21   U.IM.N}
0DB0:98 08 21 6C 0E CD F7 FE   ..!l.Mw
0DB8:01 80 00 21 25 1C 04 23   ...!%..
0DC0:7E FE FF 28 06 B9 CC D1   ~~.(.9L
0DC8:0D 18 F3 CD A4 08 C3 0D   ..sM$.C
0DD0:0A E5 C5 21 B4 0B CD 35   .eE!4.M
0DD8:0C FD 6E 00 FD 66 01 FD   .}n.}f.
0DE0:23 FD 23 CD EA 08 C1 E1   #}#Mj.A
0DE8:C9 CD 9E CE 21 2F 0E CD   IM.N!/.
0DF0:F7 FE 21 62 0E 18 2F CD   w~!b../
0DF8:9E CE 21 16 0C 11 75 08   .N!...u
0E00:01 1F 00 ED B0 3A 47 15   ...m0:G
0E08:32 94 08 3A 86 15 32 95   2..:..2
0E10:08 3A D2 0F 32 96 08 3A   .:R.2..
0E18:EF 0E 32 97 08 21 2F 0E   o.2..!/
0E20:CD F7 FE 21 58 0E CD F7   Mw~!X.M
0E28:FE CD A4 08 C3 65 09 50   ~M$.Ce.
0E30:6F 75 72 20 72 65 70 61   our rep
0E38:72 74 69 72 20 64 75 20   rtir du
```

FIGURE 35 : SUITE

```
0E40:6D 9B 2D 6D 65 20 65 6E   m.-me e
0E48:64 72 6F 69 74 20 66 61   droit f
0E50:69 72 65 20 3A 20 20 00   ire :
0E58:45 58 45 43 26 48 38 31   EXEC&H8
0E60:35 00 45 58 45 43 26 48   5.EXEC&
0E68:38 33 38 00 56 6F 75 73   838.Vou
0E70:20 70 6F 73 73 65 64 65    possed
0E78:7A 20 3A 00 00 44 65 76   z :..De
0E80:61 6E 74 20 75 6E 65 20   ant une
0E88:6D 61 69 73 6F 6E 20 2A   maison
0E90:00 52 9C 64 75 69 74 20   .R.duit
0E98:20 2A 20 45 00 44 9C 62    * E.D.
0EA0:75 74 20 63 6F 75 6C 6F   ut coul
0EA8:69 72 20 2A 20 4E 45 57   ir * NE
0EB0:00 4C 65 20 73 61 6C 6F   .Le sal
0EB8:6E 20 2A 20 4E 45 57 00   n * NEW
0EC0:55 6E 20 62 75 72 65 61   Un bure
0EC8:75 20 2A 20 57 00 55 6E   u * W.U
0ED0:20 70 61 6C 69 65 72 20    palier
0ED8:2A 20 4E 00 4D 69 6C 69   * N.Mil
0EE0:65 75 20 63 6F 75 6C 6F   eu coul
0EE8:69 72 20 2A 4E 53 45 57   ir *NSE
0EF0:00 4C 61 20 63 75 69 73   .La cui
0EF8:69 6E 65 20 2A 20 4E 53   ine * N
0F00:00 50 75 69 74 73 20 73   .Puits
0F08:61 6E 73 20 66 6F 6E 64   ans fon
0F10:20 2A 00 55 6E 65 20 63    *.Une
0F18:68 61 6D 62 72 65 20 2A   hambre
0F20:20 53 00 46 6F 6E 64 20    S.Fond
0F28:64 75 20 63 6F 75 6C 6F   du coul
0F30:69 72 20 2A 20 4E 53 00   ir * NS
0F38:54 6F 62 6F 67 67 61 6E   Tobogga
0F40:20 2A 20 45 00 43 61 76    * E.Ca
0F48:65 20 2A 20 4E 53 45 57   e * NSE
0F50:00 55 6E 65 20 63 68 61   .Une ch
0F58:6D 62 72 65 20 2A 00 55   mbre *.
0F60:6E 20 72 9C 64 75 69 74   n r.dui
0F68:20 2A 20 53 00 43 61 76    * S.Ca
0F70:65 20 2A 20 4E 53 45 00   e * NSE
0F78:43 61 76 65 20 2A 20 4E   Cave *
0F80:53 45 00 46 75 6D 6F 69   SE.Fumo
0F88:72 20 2A 20 4E 00 42 69   r * N.B
0F90:62 6C 69 6F 74 68 65 71   bliothe
0F98:75 65 20 2A 20 45 57 00   ue * EW
0FA0:43 61 76 65 20 2A 20 4E   Cave *
0FA8:53 45 57 00 43 61 76 65   SEW.Cav
0FB0:20 2A 20 4E 53 45 57 00    * NSEW
0FB8:52 9C 73 65 72 76 65 20   R.serve
0FC0:2A 20 53 45 00 55 6E 20   * SE.Un
0FC8:62 75 72 65 61 75 20 2A   bureau
```

FIGURE 35 : SUITE

```
0FD0:20 53 14 00 50 65 74 69    S..Pet
0FD8:74 20 74 75 6E 6E 65 6C   t tunn
0FE0:20 2A 20 45 57 00 56 6F    * EW.
0FE8:75 73 20 9B 74 65 73 20   us .te
0FF0:73 6F 72 74 69 20 2A 00   sorti
0FF8:4C 61 20 62 6F 6D 62 65   La bomb
1000:20 65 78 70 6C 6F 73 65    explo
1008:20 2A 00 FF 00 50 6F 72    *...P
1010:74 65 20 61 76 65 63 20   te ave
1018:70 6C 61 71 75 65 00 50   plaque
1020:6F 73 74 65 20 72 61 64   oste r
1028:69 6F 00 4C 61 20 70 6F   io.La p
1030:72 74 65 20 65 73 74 20   rte est
1038:62 6C 6F 71 75 9C 65 00   bloqu.e
1040:42 75 66 66 65 74 20 61   Buffet
1048:76 65 63 20 70 61 69 6E   vec pa
1050:00 43 6F 66 66 72 65 2D   .Coffre
1058:66 6F 72 74 20 66 65 72   fort f
1060:6D 9C 00 45 73 63 61 6C   m..Esc
1068:69 65 72 20 9C 63 72 6F   ier .c
1070:75 6C 9C 00 45 73 63 61   ul..Es
1078:6C 69 65 72 20 62 72 61   lier b
1080:6E 6C 61 6E 74 20 57 00   nlant
1088:54 61 62 6C 65 2C 20 4D   Table,
1090:61 67 6E 9C 74 6F 00 41   agn.to
1098:48 48 48 48 2E 2E 2E 2E   HHHH..
10A0:2E 2E 00 55 6E 65 20 70   ...Une
10A8:6F 72 74 65 20 61 75 20   orte a
10B0:4E 00 54 9C 6C BC 70 68   N.T.l<
10B8:6F 6E 65 00 43 61 20 67   one.Ca
10C0:6C 69 73 73 65 00 49 6C   lisse.
10C8:20 66 61 69 74 20 66 72    fait
10D0:6F 69 64 00 55 6E 65 20   oid.Un
10D8:70 6F 72 74 65 20 66 65   porte
10E0:72 6D 9C 65 00 50 6F 72   rm.e.P
10E8:74 65 20 63 6C 6F 73 65   te clo
10F0:20 61 75 20 4E 00 49 6C    au N.
10F8:20 66 61 69 74 20 66 72    fait
1100:6F 69 64 00 49 6C 20 66   oid.Il
1108:61 69 74 20 66 72 6F 69   ait fr
1110:64 00 4C 61 20 62 6F 6D   d.La b
1118:62 65 20 65 73 74 20 61   be est
1120:6D 6F 72 63 9C 65 00 50   morc.e
1128:6F 72 74 65 20 61 75 20   orte a
1130:53 20 2D 20 50 61 70 69   S - Pa
1138:65 72 73 00 49 6C 20 66   ers.Il
1140:61 69 74 20 66 F2 6F 69   ait fr
1148:64 00 49 6C 20 66 61 69   d.Il f
1150:74 20 66 72 6F 69 64 00   t froi
1158:41 67 69 73 73 65 7A 20   Agisse
```

FIGURE 35 : SUITE

```
1160:76 69 74 65 00 43 68 61   vite.Ch
1168:69 73 65 73 20 2D 20 74   ises -
1170:61 62 6C 65 61 75 00 42   ableau.B
1178:72 72 72 20 21 21 21 00   rrr !!!.
1180:46 9C 6C 69 63 69 74 61   F.licita
1188:74 69 6F 6E 73 20 21 00   tions !
1190:42 4F 4F 4F 4F 55 4D 20   BOOOOUM
1198:21 21 21 21 00 49 6C 20   !!!!.Il
11A0:66 61 69 74 20 6E 6F 69   fait no
11A8:72 00 50 6F 72 74 65 20   r.Porte
11B0:6F 75 76 65 72 74 65 00   ouverte
11B8:43 6F 66 66 72 65 2D 66   Coffre-
11C0:6F 72 74 20 6F 75 76 65   ort ouve
11C8:72 74 00 43 6F 72 64 65   rt.Corde
11D0:20 70 65 6E 64 75 65 20    pendue
11D8:65 6E 20 57 00 2B 51 1B   en W.+Q
11E0:3C 43 4C 45 46 81 54 4F   <CLEF.TO
11E8:52 43 82 50 41 50 49 83   RC.PAPI
11F0:43 41 53 53 84 43 4F 52   CASS.COR
11F8:44 85 43 41 46 45 86 4D   D.CAFE.
1200:41 52 54 87 43 4C 45 53   ART.CLE
1208:88 42 55 52 49 89 4C 49   .BURI.L
1210:56 52 8A 54 45 4E 41 8B   VR.TENA
1218:50 49 4C 45 8C 50 4F 4D   PILE.PO
1220:4D 8D 53 45 52 52 8E 50   M.SERR.
1228:4F 52 54 8F 43 4F 46 46   ORT.COF
1230:90 43 41 56 45 91 31 39   .CAVE.1
1238:34 32 92 42 4F 4D 42 93   42.BOMB
1240:54 41 42 4C 94 4D 41 47   TABL.MA
1248:4E 95 43 4F 55 4C 96 43   N.COUL.
1250:4C 4F 55 97 50 4C 41 51   LOU.PLA
1258:98 50 41 49 4E 99 52 41   .PAIN.R
1260:44 49 9A 54 45 4C 45 9B   DI.TELE
1268:00 93 12 FA 12 2F 13 4E   ...z./.
1270:13 63 13 72 13 98 13 D3   .c.r...
1278:13 DD 13 F8 13 07 14 31   .].x...
1280:14 36 14 40 14 4E 14 60   .6.@.N.
1288:14 84 14 89 14 CC 14 DF   .....L.
1290:14 13 15 FE 0F 28 2B FE   ...~.(+
1298:0E 28 0B FE 10 C2 E3 14   .(.~.Bc
12A0:11 7F 16 C3 74 0A 3A 1D   ...Ct.:
12A8:0C FE 17 C2 71 0A 3E 06   .~.Bq.>.
12B0:CD C7 17 20 F6 3E 18 32   MG. v>.
12B8:86 15 3E 45 32 D2 0F C3   ..>E2R.(
12C0:7D 0A 3A 16 0C FE 01 20   }.:..~.
12C8:08 3E 03 32 16 0C C3 7D   .>.2..C.
12D0:0A FE 0A 20 04 3E 0E 18   .~. .>..
12D8:F2 FE 0E 20 10 3E 01 CD   r~. .>.
12E0:C7 17 28 05 11 A1 16 18   G.(..!..
12E8:BA 3E 0A 18 DE FE 13 20   :>..^~.
```

FIGURE 35 : SUITE

```
12F0:BA 3E 01 32 1C 0C 3E 0F
12F8:18 D1 FE 0E D2 71 0A CD
1300:BF 17 3A 16 0C BE C2 71
1308:0A E5 21 26 0C 7E FE FF
1310:28 08 23 FE 80 20 F6 14
1318:18 F3 E1 7A FE 05 38 09
1320:11 95 15 CD CF 0A C3 AB
1328:0D 3E 80 77 C3 7D 0A FE
1330:0E D2 71 0A CD C7 17 28
1338:06 11 A6 15 C3 74 0A 3A
1340:31 0B FE 05 3A 16 0C 20
1348:E2 32 21 0C 18 DD FE 05
1350:C2 6C 13 3A 21 0C FE 06
1358:C2 71 0A 3E 07 32 16 0C
1360:C3 7D 0A FE 06 20 05 11
1368:B9 15 18 03 11 C3 15 C3
1370:74 0A FE 03 20 0B CD C7
1378:17 C2 39 13 11 D4 16 18
1380:EE FE 0A 20 0A CD C7 17
1388:20 EF 11 DA 15 18 E0 FE
1390:18 20 D9 11 6A 17 18 D7
1398:FE 14 20 18 3A 1E 0C FE
13A0:17 20 0C 32 1D 0C 32 33
13A8:0C 11 44 17 C3 80 0A 11
13B0:37 17 18 BB FE 17 C2 E3
13B8:14 3E 03 CD C7 17 28 05
13C0:11 57 17 18 AA 3A 16 0C
13C8:FE 17 C2 71 0A 32 1E 0C
13D0:C3 7D 0A FE 16 C2 6C 13
13D8:11 B5 16 18 22 FE 14 20
13E0:05 11 F4 16 18 19 FE 03
13E8:28 09 FE 0A 28 05 11 E4
13F0:16 18 0C 11 05 17 18 07
13F8:FE 0F 20 06 11 20 16 C3
1400:74 0A 11 F6 15 18 F8 FE
1408:02 20 1C 3A 1B 0C FE 02
1410:20 05 11 2D 16 18 E8 FE
1418:01 28 05 11 1B 16 18 DF
1420:3C 32 1B 0C C3 7D 0A FE
1428:1A C2 6C 13 11 91 16 18
1430:CE 11 E9 15 38 C9 FE 0F
1438:C2 E3 14 11 26 17 18 BF
1440:FE 19 20 05 11 7A 17 18
1448:B6 FE 0D C3 65 13 FE 05
1450:C2 6C 13 3A 21 0C FE 06
1458:C2 02 14 32 16 0C 18 5E
1460:FE 0F 28 05 11 12 17 18
1468:37 3A 16 0C FE 0F C2 7D
1470:0A 3A 1C 0C FE 01 28 05
1478:11 E5 10 18 23 3E 13 32
```

FIGURE 35 : SUITE

```
1480:16 0C 18 32 11 12 17 18   ...2...
1488:17 FE 13 20 06 11 C5 16   .~. ..E
1490:C3 74 0A FE 04 20 0C CD   Ct.~. .|
1498:C7 17 C2 39 13 C3 5E 18   G.B9.C^.
14A0:C3 74 0A FE 1A 20 12 3A   Ct.~. .
14A8:1F 0C FE 02 C2 1B 14 DD   ..~.B..]
14B0:21 F3 17 CD BB 08 C3 7D   !s.M;.C}
14B8:0A FE 15 20 05 11 AE 17   .~. ....
14C0:18 DE FE 1B C2 6C 13 11   .^~.Bl..
14C8:9E 17 18 D4 FE 02 C2 6C   ...T~.B
14D0:13 3A 1B 0C FE 02 C2 12   .:..~.B.
14D8:14 3D 32 1B 0C 18 D7 FE   .=2...W~
14E0:0C 28 05 11 3D 16 18 B8   .(..=..8
14E8:3E 0C CD C7 17 C2 39 13   >.MG.B9
14F0:3A 16 0C FE 02 20 05 32   :..~. .
14F8:1F 0C 18 11 3E 02 CD C7   ....>.M
1500:17 28 05 11 4C 16 18 98   .(..L..
1508:3E 01 32 1B 0C AF 32 31   >.2../2
1510:0C 18 A3 FE 12 20 CC 3A   ..#~. L
1518:16 0C FE 05 C2 71 0A 3E   ..~.Bq.
1520:01 32 19 0C 3E 05 32 30   .2..>.2
1528:0C C3 7D 0A 00 00 00 00   .C}....
1530:00 00 03 00 07 00 04 02   .......
1538:08 00 05 03 00 00 00 04   .......
1540:0A 00 00 00 0B 03 08 06   .......
1548:0C 04 00 00 00 00 00 00   .......
1550:00 06 00 00 0F 07 00 00   .......
1558:00 00 0D 00 11 09 0D 0D   .......
1560:00 00 00 00 00 0B 00 00   .......
1568:14 10 11 00 15 0D 11 10   .......
1570:16 00 00 00 00 00 14 12   .......
1578:14 10 15 13 15 11 00 14   .......
1580:00 12 17 00 00 13 00 00   .......
1588:00 00 19 17 00 00 00 00   .......
1590:00 00 00 00 FF 56 6F 75   .....Vo
1598:73 20 70 6F 72 74 65 7A   s porte
15A0:22 74 72 6F 70 00 56 6F    trop.V
15A8:75 73 20 6E 65 20 6C 27   us ne l
15B0:61 76 65 7A 20 70 61 73   avez pa
15B8:00 43 27 65 73 74 20 62   .C'est
15C0:6F 6E 00 51 75 65 6C 6C   on.Quel
15C8:65 20 69 64 9C 65 00 46   e id.e.
15D0:41 49 52 65 20 31 39 34   AIRe 19
15D8:32 00 43 27 65 73 74 20   2.C'est
15E0:75 6E 20 70 6F 6C 61 72   un pola
15E8:00 47 72 61 6E 64 65 20   .Grande
15F0:62 72 75 74 65 00 4F 68   brute.O
```

FIGURE 35 : SUITE

```
15F8:20 68 69 73 73 65 2E 00
1600:45 73 73 61 79 65 72 20
1608:4F 55 56 52 65 00 43 27
1610:65 73 74 20 62 6C 6F 71
1618:75 9C 00 49 6C 20 66 61
1620:75 74 20 64 65 73 20 70
1628:69 6C 65 73 00 43 27 65
1630:73 74 20 64 9C 6A 61 20
1638:66 61 69 74 00 4A 65 20
1640:6E 65 20 73 61 69 73 20
1648:70 61 73 00 4F 75 20 63
1650:65 6C 61 20 3F 00 0A 20
1658:42 72 61 76 6F 2E 2E 2E
1660:20 41 20 62 69 65 6E 74
1668:6F 74 00 0A 20 50 61 73
1670:20 64 65 20 63 68 61 6E
1678:63 65 2E 2E 2E 20 00 46
1680:41 49 52 65 20 63 6F 6D
1688:62 69 6E 61 69 73 6F 6E
1690:00 45 73 73 61 79 65 7A
1698:20 45 43 4F 55 74 65 72
16A0:00 41 76 6F 69 72 20 6C
16A8:61 20 62 6F 6E 6E 65 20
16B0:63 6C 65 46 00 54 72 6F
16B8:70 20 64 61 6E 67 65 75
16C0:72 65 75 78 00 54 69 63
16C8:20 54 61 63 20 54 69 63
16D0:2E 2E 2E 00 45 4E 4C 45
16D8:76 65 72 20 54 41 42 4C
16E0:65 61 75 00 54 6F 75 74
16E8:20 65 73 74 20 6E 6F 72
16F0:6D 61 6C 00 43 27 65 73
16F8:74 20 75 6E 65 20 63 72
1700:6F 75 74 65 00 45 73 73
1708:61 79 65 7A 20 4C 49 52
1710:45 00 55 74 69 6C 69 73
1718:65 7A 20 4E 2C 53 2C 45
1720:20 6F 75 20 57 00 45 6C
1728:6C 65 20 65 73 74 20 63
1730:6F 69 6E 63 9C 65 00 49
1738:6C 20 65 73 74 20 63 6C
1740:6F 75 9C 00 49 6C 20 79
1748:20 61 20 75 6E 65 20 73
1750:65 72 72 75 72 65 00 49
1758:6C 20 66 61 75 74 20 64
1760:65 73 20 6F 75 74 69 6C
1768:73 00 45 6E 74 72 9C 65
1770:20 67 72 61 74 75 69 74
1778:65 00 49 6C 20 65 73 74
1780:20 72 61 73 73 69 73 00
```

FIGURE 35 : SUITE

1788:0A 56 6F 75 73 20 65 6E	.Vous e
1790:74 72 65 7A 20 64 61 6E	trez da
1798:73 20 2E 2E 2E 00 50 61	s ....P
17A0:73 20 64 65 20 74 6F [illegible]	s de ta
17A8:6[illegible] 6C 69 74 9C 00 45 4[illegible]	alit..E
17B0:4F 55 74 65 72 20 43 41	OUter C
17B8:[illegible] 53 65 74 74 E5 00 21	SSette.
17C0:25 0C 5F 16 00 19 C9	%._...I
17C8:6F 12 3E 60 nE C9 CD 4E	?.>.>IM
17D0:CE 21 56 16 CD F7 FE 3D	N!V.Mw~
17D8:21 22 18 CD BB 08 CD 64	!".M;.r
17E0:08 C3 65 09 CD 9E CE [illegible]	.Ce.M.N
17E8:6B 16 CD F7 FE DD 21 4[illegible]	k.Mw~]!
17F0:18 18 E8 11 04 11 04 11	..h.....
17F8:24 0D 10 20 10 0F 04 0F	....
1800:04 0F 04 0C 10 [illegible] 10	....
1808:[illegible] 11 04 11 04 11 04 [illegible]	....
1810:24 12 04 12 04 12 04 11	....
1818:24 19 04 19 04 19 04 16	....
1820:17 FF 14 0C [illegible] 24 16 46	....
1828:14 08 00 08 14 0C 14 04	....
1830:16 08 14 08 00 08 16 08	....
1838:14 08 12 06 [illegible] 06 0F 12	....
1840:2C 2[illegible] 0C 0[illegible] FF 2C 10 0[illegible]	....
1848:0C 2C 24 2C 0C 10 04 10	....
1850:0C 0F 04 0F 0C 0D 04 0D	....
1858:06 2C 08 0[illegible] 10 FF 3A 16	.......:
1860:0C FE 08 C2 [illegible] 0n 11 [illegible]	.~.Bq..
1868:15 C3 74 26 00 00 00 00	.Ct....

FIGURE 35 : FIN

```
5 REM *** CHARGEUR - VERSION LCD **
10 CLEAR50,&H7FF
20 INIT#1,"CASI:"
30 INPUT#1,N$,D,F
40 MOTOR
50 PRINT"Trouv :";N$
60 FORI=D-1TOF
70 POKEI,INP(#1)
80 NEXT:MOTOR
90 END
100 CLEAR50,&H9FF
110 D=&H800:F=&H186C
120 N$="AVENT2":INIT#1,"CASO:"
130 INPUT"Magnto OK";T$
140 PRINT#1,N$,D,F:MOTOR
150 FORI=1TO1800:NEXT
160 FORI=DTOF:OUT#1,PEEK(I)
170 NEXT:MOTOR
180 END
```

```
5 REM *** ENTREUR DE CODES - LCD **
10 CLEAR50,&H7FF:A=&H800
20 PRINTHEX$(A);"  :  ";:INPUT C$
30 V=VAL("&H"+C$):POKE A,V
40 A=A+1:IF A>&H186C THEN PRINT"
..."":BEEP2,3:END
50 GOTO20
```

FIGURE 36 : CHARGEURS

```
0800:CD 91 08 DC 21 2C 18 CD   M..]!!.!
0808:44 09 21 DB 08 11 FE 08   D.!L..~
0810:01 23 00 ED B0 21 FE 08   .#.m0!~
0818:11 8C 0C 01 1F 00 ED B0   .......m
0820:3A 1D 09 32 D0 15 3A 1E   :..2P.:
0828:09 32 0F 16 3A 20 09 32   .2..: .
0830:6B 0F 3A 1F 09 32 4E 10   k.:..2N
0838:F3 21 89 09 22 3D 00 FB   s!.."=.
0840:3E 80 32 00 80 3A 00 80   >.2..:.
0848:FE 80 C2 9B 08 3E 00 4F   ~.B..>.
0850:3E 01 57 5F CD 09 AB 3E   >.W_M.+
0858:01 32 E5 04 3E 02 32 E6   .2e.>.2
0860:04 CD 9E CE 21 11 18 CD   .M.N!..
0868:F7 FE CD 80 09 CD 04 0A   w~M..M.
0870:CD 9E CE 21 01 10 11 6D   M.N!...
0878:0A CD 6A 09 21 02 01 CD   .Mj.!..
0880:73 09 21 07 01 CD 73 09   s.!..Ms
0888:21 08 01 22 BC 09 C3 29   !.."<.C
0890:0A CD 9E CE 21 BA 08 CD   .M.N!:.
0898:F7 FE C9 21 A7 08 CD F7   w~I!'.M
08A0:FE CD 2D 09 C3 F9 09 0A   ~M-.Cy.
08A8:0D 58 2D 37 32 30 20 6E   .X-720
08B0:6F 6E 20 61 63 74 69 76   on acti
08B8:65 00 0A 20 22 20 4C 65   e..   !
08C0:20 50 69 65 67 65 0A 0D    Piege.
08C8:56 65 72 73 69 6F 6E 20   Version
08D0:70 6F 75 72 20 58 2D 37   pour X-
08D8:32 30 00 01 00 00 00 00   20.....
08E0:00 00 00 00 00 00 00 00   .......
08E8:00 00 00 0F 04 0A 17 02   .......
08F0:08 08 0E 0F 05 00 02 16   .......
08F8:00 FF 06 00 14 57 0C 01   .....W.
0900:00 00 00 02 00 00 00 00   .......
0908:02 00 01 01 00 00 0A 80   .......
0910:0A 17 06 08 08 80 0F 05   .......
0918:00 02 16 00 FF 00 00 14   .......
0920:20 04 01 04 10 05 01 05    ......
0928:10 0E 01 06 10 CD BD C0   .....M=
0930:AF CD 0A C9 28 FA C9 DD   /M.I[zI
0938:21 3F 09 CD 44 09 C9 17   !?.MD.I
0940:02 20 02 FF DD 7E 00 FE   . ..]~.
0948:FF C8 5F DD 46 01 16 00   .H_]F..
0950:4A DD 23 DD 23 21 61 09   J]#]#!a
0958:E5 C5 D5 FD E1 E1 CD DF   eEU}aaM
0960:C2 2A 0E 00 7C F5 20 F9   B*..|u
0968:18 DA 22 B8 00 D5 E1 CD   .Z"8.Ua
0970:F7 FE C9 22 B8 00 06 20   w~I"8..
0978:3E 2D CD 9F 00 10 F9 C9   >-M...y
0980:01 20 00 0B 78 B1 20 FB   ....x1
0988:C9 D9 08 DB F1 FE 11 C2   IY.[q~.
```

FIGURE 37 : LE PIEGE (VIDEO)

```
0990:9B C7 08 D9 ED 57 E2 C3
0998:C3 CD BD C0 3A 2B 00 06
09A0:08 B0 32 2B 00 3E 0C D3
09A8:BB CD A2 00 C3 CF C3 CD
09B0:DC 09 CD 9E CE 21 03 01
09B8:22 B8 00 C3 10 01 3E 01
09C0:32 BB 00 3E 10 32 BC 00
09C8:32 BD 00 C9 3E 08 32 BB
09D0:00 3E 10 32 BC 00 3E 09
09D8:32 BD 00 C9 3E 03 32 BB
09E0:00 3E 04 32 BD 00 3E 06
09E8:32 BC 00 C9 CD BE 09 AF
09F0:32 D1 00 4F 57 5F CD 09
09F8:AB F3 21 99 C7 22 3D 00
0A00:FB C3 28 C5 CD 37 09 CD
0A08:BD C0 21 07 01 CD 73 09
0A10:21 09 01 CD 73 09 21 08
0A18:08 22 B8 00 0E 0C 3E 06
0A20:CD 0A C9 C0 C5 01 00 35
0A28:CD 83 09 C1 79 FE 0B 28
0A30:0D CD 5F 0A 0C 79 FE 21
0A38:20 E4 0E 01 18 E0 C5 3E
0A40:21 91 67 41 22 B8 00 21
0A48:6D 0A 7E CD 9F 00 23 10
0A50:F9 11 08 01 ED 53 B8 00
0A58:CD F7 FE C1 0C 18 BF 3E
0A60:21 91 67 22 B8 00 21 6D
0A68:0A CD F7 FE C9 2A 4C 45
0A70:20 50 49 45 47 45 2A 20
0A78:00 CD C5 0C CD 5B 0D CD
0A80:F6 0D 3A 8C 0C FE 1A CA
0A88:6D 18 FE 09 28 F9 FE 19
0A90:CA 57 18 3A 99 0C FE 01
0A98:20 57 3A 6E 02 FE 49 CA
0AA0:27 0E FE 51 CA 65 0E FE
0AA8:50 CA 73 0E 0E 00 FE 4E
0AB0:28 17 1C FE 53 28 12 0C
0AB8:FE 45 28 0D 0C FE 57 28
0AC0:08 11 7C 0B CD 3B 0B 18
0AC8:B6 3A 8C 0C 4F 11 04 00
0AD0:21 F1 15 39 05 20 FC 59
0AD8:19 AF BE 20 08 11 8A 0B
0AE0:CD 3B 0B 18 9A 7E 32 8C
0AE8:0C 11 A2 0B CD 3B 0B 18
0AF0:88 CD 54 0B FE 00 20 05
0AF8:11 7C 0B 18 E3 32 A6 0B
0B00:1A FE 00 28 F3 FE 20 28
0B08:05 13 18 F4 13 EB 11 6E
0B10:02 01 05 00 ED B0 21 58
0B18:12 CD 57 0B FE 00 20 05
```

FIGURE 37 : SUITE

```
0B20:11 81 0B 18 BB 32 A7 0B   ....;2'
0B28:3A A6 0B 87 26 00 6F 11   :&..&.o
0B30:E3 12 19 5E 23 56 EB 3A   c..^#Vk
0B38:A7 0B E9 CD CC 09 2A BC   '.iML.*
0B40:09 CD 6A 09 CD 37 09 11   .Mj.M7.
0B48:79 0B CD 6D 09 2A B8 00   y.Mm.*8
0B50:22 BC 09 C9 21 A3 0B 01   "<.I!#.
0B58:05 00 09 AF BE C6 E5 DD   .../>He
0B60:E1 11 6E 02 01 04 00 1A   a.n....
0B68:ED A1 20 08 13 EA 67 0B   m! ..j9
0B70:7E EE 80 C9 DD E5 E1 18   ~n.I]ea
0B78:DE 0A 0D 00 3F 3F 3F 3F   ^...???
0B80:00 2A 2A 2A 2A 3F 3F 3F   .****??
0B88:3F 00 49 6D 70 6F 73 73   ?.Impos
0B90:69 62 6C 65 00 31 34 32   ible.14
0B98:30 39 37 32 31 38 31 30   0972181
0BA0:38 39 4F 2E 4B 00 12 04   890.K..
0BA8:4F 55 56 52 81 50 52 45   OUVR.PR
0BB0:4E 82 50 4F 53 45 83 4C   N.POSE.
0BB8:41 43 48 83 44 45 53 43   ACH.DES
0BC0:84 42 4F 49 52 85 4C 49   .BOIR.L
0BC8:52 45 86 4F 54 45 52 87   RE.OTER
0BD0:45 5E 4C 45 87 53 41 55   E^LE.SA
0BD8:54 88 52 45 47 41 89 45   T.REGA.
0BE0:58 41 4D 89 54 49 52 45   XAM.TIR
0BE8:8A 50 4F 55 53 8A 41 4C   .POUS.A
0BF0:4C 55 8B 46 52 41 50 9C   LU.FRAP
0BF8:43 41 53 53 8C 46 45 52   CASS.FE
0C00:4D 8D 4D 41 4E 47 8E 4D   M.MANG.
0C08:4F 4E 54 8F 41 4C 4C 45   ONT.ALL
0C10:90 45 4E 54 52 91 45 43   .ENTR.E
0C18:4F 55 92 45 54 45 49 93   OU.ETEI
0C20:4D 45 54 54 94 46 41 49   METT.FA
0C28:52 95 00 00 43 4C 45 46   R...CLE
0C30:00 54 4F 52 43 48 45 00   .TORCHE
0C38:50 41 50 49 45 52 00 43   PAPIER.
0C40:41 53 53 45 54 54 45 00   ASSETTE
0C48:43 4F 52 44 45 00 43 41   CORDE.C
0C50:46 45 00 4D 41 52 54 45   FE.MARTI
0C58:41 55 00 43 4C 45 53 00   AU.CLES
0C60:42 55 52 49 4E 00 4C 49   BURIN.L
0C68:56 52 45 00 54 45 4E 41   VRE.TEN
0C70:49 4C 4C 45 53 00 50 49   ILLES.P
0C78:4C 45 53 00 50 4F 4D 4D   LES.POM
0C80:45 53 00 53 45 52 52 55   ES.SERR
0C88:52 45 00 FF 1A 0F 20 00   RE....
0C90:00 00 01 00 00 00 02 06   .......
0C98:01 01 00 12 0A 00 0A 80   .......
0CA0:06 08 80 80 0F 05 00 00   .......
0CA8:16 00 FF 23 7E FE 00 20   ...#~~.
```

FIGURE 37 : SUITE

```
0CB0:FA 05 20 F7 23 EB C9 3A   z. w#k
0CB8:9A 0C FE 02 C8 3C 32 9A   ..~.H<
0CC0:0C CD 4D 0E C9 3A 9B 0C   .MM.I:
0CC8:FE 00 28 11 3A 8E 0C 3C   ~.(.:.
0CD0:32 8E 0C FE 20 20 06 3E   2..~
0CD8:1A 32 8C 0C C9 3A 90 0C   .2..I:
0CE0:FE 00 28 0B 3A 91 0C FE   ~.(.:.
0CE8:02 28 04 3E 09 18 EA 3A   .(.>..
0CF0:91 0C FE 02 20 22 3A 8D   ..~. "
0CF8:0C 3C 32 8D 0C FE 0F 20   .<2..~
0D00:17 CD AF 09 21 4B 0D CD   .M/.!K
0D08:F7 FE 21 9D 0C AF 77 72   w~!../
0D10:91 0C CD 80 09 C3 80 09   ..M..C
0D18:3A 8C 0C FE 06 20 13 3A   :..~.
0D20:B8 0C FE 00 C0 32 D0 15   8.~.@2
0D28:3C 32 98 0C 3E 20 32 6B   <2..> 2
0D30:0F C9 FE 12 20 04 32 9B   .I~. .2
0D38:0C C9 FE 0E C0 3A 96 0C   .I~.@:
0D40:3C 32 96 0C FE 03 C0 E1   <2..~.@
0D48:C3 6D 18 0A 50 69 6C 65   Cm..Pil
0D50:73 20 9C 70 75 69 73 BC   s .puis
0D58:65 73 00 CD AF 09 3A 8C   es.M/.:
0D60:0C 47 21 F7 0E CD AB 0C   .G!w.M+
0D68:21 03 01 CD 6A 09 AF FD   !..Mj./
0D70:21 21 09 32 9A 0C 47 3A   !!.2..G
0D78:8C 0C 4F 21 8C 0C 7E FE   ..O!..~
0D80:FF 28 08 04 23 B9 CC B7   .(..#9L
0D88:0C 18 F3 3A 8C 0C 47 AF   ..s:..G
0D90:32 90 0C 78 CD A4 0D 21   2..xM$.
0D98:87 10 CD AB 0C 21 05 01   ..M+.!.
0DA0:CD 6A 09 C9 FE 05 20 09   Mj.I~.
0DA8:3A 8F 0C FE 00 C8 06 1D   :..~.H.
0DB0:C9 FE 0D 28 12 FE 10 28   I~.(.~.
0DB8:0E FE 11 28 0A FE 14 28   .~.(.~.
0DC0:06 FE 15 28 02 18 0E 3E   .~.(...
0DC8:01 32 90 0C 3A 91 0C FE   .2..:..
0DD0:02 C8 06 1B C9 FE 0F 20   .H..I~.
0DD8:09 3A 92 0C FE 01 C0 06   .:..~.@
0DE0:1C C9 FE 07 C0 3A 98 0C   .I~.@:.
0DE8:FE 00 C8 06 06 3A 97 0C   ~.H..:.
0DF0:FE 06 C0 06 1E C9 CD CC   ~.@..IM
0DF8:09 CD BD C0 2A BC 09 22   .M=@*<.
0E00:F8 00 CD F2 EB 23 11 6E   x.Mrk#.
0E08:02 0E 00 7E 12 B7 28 08   ...~.7(
0E10:23 13 0C 79 FE 14 38 F3   #..y~.8
0E18:79 2A B8 00 22 BC 09 FE   y*8."<.
0E20:00 28 D6 32 99 0C C9 CD   .(V2..I
0E28:AF 09 FD 21 21 09 21 E8   /.}!!.!
0E30:0E CD F7 FE 01 80 00 21   .Mw~...
0E38:9B 0C 04 23 7E FE FF 28   ...#~~.
```

FIGURE 37 : SUITE

```
0E40:06 B9 CC 4D 0E 18 F3 CD    .9LM..s
0E48:2D 09 C3 79 0A E5 C5 21    -.Cy.eE
0E50:2A 0C CD AB 0C FD 6E 00    *.M+.}n
0E58:FD 66 01 FD 23 FD 23 CD    }f.}#}#
0E60:6A 09 C1 E1 C9 CD AF 09    j.AaIM/
0E68:21 AB 0E CD F7 FE 21 DE    º+.Mw~º
0E70:0E 38 2F CD AF 09 21 8C    .8/M/.º
0E78:0C 11 FE 06 01 1F 00 EC    ..~....
0E80:B0 3A D0 15 32 1D 09 3A    0:P.2..
0E88:0F 16 32 1E 09 3A 4E 10    ..2..:N
0E90:32 1F 09 3A 6B 0F 32 20    2..:k.2
0E98:09 21 AB 0E CD F7 FE 21    .º+.Mw~
0EA0:D4 0E CD F7 FE CD 2D 09    T.Mw~M-
0EA8:C3 EC 09 50 6F 75 72 20    Cl.Pour
0EB0:72 65 70 61 72 74 69 72    reparti
0EB8:20 64 75 20 6D 9B 6D 65     du m.m
0EC0:20 65 6E 64 72 6F 69 74     endroi
0EC8:0A 0D 66 61 69 72 65 20    ..faire
0ED0:3A 20 20 00 45 58 45 43    :   .EXE
0ED8:26 48 38 31 35 00 45 58    &H815.E
0EE0:45 43 26 48 78 33 38 00    EC&Hx38
0EE8:56 6F 75 73 20 70 6F 73    Vous po
0EF0:73 65 64 65 7A 20 3A 00    sedez :
0EF8:00 44 65 76 61 6E 74 20    .Devant
0F00:75 6E 65 20 6D 61 69 73    une mai
0F08:6F 6E 20 2A 00 52 9C 64    on *.R.
0F10:75 69 74 20 20 2A 20 45    uit  *
0F18:00 44 9C 62 75 74 20 63    .D.but
0F20:6F 75 6C 6F 69 72 20 2A    ouloir
0F28:20 4E 45 57 00 4C 65 20     NEW.Le
0F30:73 61 6C 6F 6E 20 2A 20    salon *
0F38:4E 45 57 00 55 6E 20 62    NEW.Un
0F40:75 72 65 61 75 20 2A 20    ureau *
0F48:57 00 55 6E 20 70 61 6C    W.Un pa
0F50:69 65 72 20 2A 20 4E 00    ier * N
0F58:4D 69 6C 69 65 75 20 63    Milieu
0F60:6F 75 6C 6F 69 72 20 2A    ouloir
0F68:4E 53 45 20 00 4C E1 20    NSE .La
0F70:63 75 69 73 69 6E 65 20    cuisine
0F78:2A 20 4E 53 00 50 75 69    * NS.Pu
0F80:74 73 20 73 61 7E 73 20    ts sa~s
0F88:66 6F 6E 64 20 2A 00 55    fond *.
0F90:6E 65 20 63 68 61 6D 62    ne cham
0F98:72 65 20 2A 20 53 00 46    re * S.
0FA0:6F 6E 64 20 64 75 20 63    ond du
0FA8:6F 75 6C 6F 69 72 20 2A    ouloir
0FB0:20 4E 53 00 54 6F 62 6F     NS.Tob
0FB8:67 67 E1 6E 20 2A 20 45    ggan *
0FC0:00 43 61 76 E5 20 2A 20    .Cave *
0FC8:4E 53 45 57 00 55 6E 65    NSEW.Un
```

FIGURE 37 : SUITE

```
0FD0:20 63 68 61 6D 62 72 65
0FD8:20 2A 00 55 6E 20 72 9C
0FE0:64 75 69 74 20 2A 20 53
0FE8:00 43 61 76 65 20 2A 20
0FF0:4E 53 45 00 43 61 76 65
0FF8:20 2A 20 4E 53 45 00 46
1000:75 6D 6F 69 72 20 2A 20
1008:4E 00 42 69 62 6C 69 6F
1010:74 68 65 F1 75 65 20 2A
1018:20 45 57 00 43 61 76 65
1020:20 2A 20 4E 53 45 57 00
1028:43 61 76 65 20 2A 20 4E
1030:53 45 57 00 52 9C 73 65
1038:72 76 65 20 2A 20 53 45
1040:00 55 6E 20 62 75 72 E5
1048:61 75 20 2A 20 53 14 00
1050:50 65 74 69 74 20 74 75
1058:6E 6E 65 6C 20 2A 20 45
1060:57 00 56 6F 75 73 20 9B
1068:74 65 73 20 F3 6F 72 74
1070:69 20 2A 00 4C 61 20 62
1078:6F 6D 62 65 20 65 78 70
1080:6C 6F 73 65 20 2A 00 FF
1088:00 50 6F 72 74 65 20 61
1090:76 65 63 20 70 6C E1 71
1098:75 65 00 50 6F 73 74 65
10A0:20 72 61 64 69 6F 00 4C
10A8:61 20 70 6F 72 74 65 20
10B0:65 73 74 20 62 6C 6F 71
10B8:75 9C 65 00 42 75 66 66
10C0:65 74 20 61 76 65 63 20
10C8:70 61 69 6E 00 43 6F 66
10D0:66 72 65 2D 66 6F 72 74
10D8:20 66 65 72 6D 9C 00 45
10E0:73 63 E1 6C 69 65 72 20
10E8:9C 63 72 6F 75 6C 9C 00
10F0:45 73 E3 61 6C 69 65 72
10F8:20 62 72 61 6E 6C 61 6E
1100:74 20 57 00 54 61 62 6C
1108:65 2C 20 4D 61 67 6E 9C
1110:74 6F 00 41 48 48 48 48
1118:3E 2E 2E 2E 2E 2E 00 55
1120:6E 65 20 70 6F 72 74 65
1128:20 61 75 20 4E 00 54 9C
1130:6C 9C 70 68 6F 6E 65 00
1138:43 61 20 67 6C 69 73 73
1140:65 00 49 6C 20 66 61 69
1148:74 20 66 72 6F 69 64 00
1150:55 6E 65 20 70 6F 72 74
1158:65 20 66 65 72 6D 9C E5
```

FIGURE 37 : SUITE

```
1160:00 50 6F 72 74 65 20 63   .Porte
1168:6C 6F 73 65 20 61 75 20   lose au
1170:4E 00 49 6C 20 66 61 69   N.Il fa
1178:74 20 66 72 6F 69 64 00   t froid
1180:49 6C 20 66 61 69 74 20   Il fait
1188:66 72 6F 69 64 00 4C 61   froid.L
1190:20 62 6F 6D 62 65 20 65    bombe
1198:73 74 20 61 6D 6F 72 63   st amor
11A0:9C 65 00 50 6F 72 74 65   .e.Port
11A8:20 61 75 20 53 20 2D 20    au S -
11B0:50 61 70 69 65 72 73 00   Papiers
11B8:49 6C 20 66 61 69 74 20   Il fait
11C0:66 72 6F 69 64 00 49 6C   froid.I
11C8:20 66 61 69 74 20 66 72    fait f
11D0:6F 69 64 00 41 67 69 73   oid.Agi
11D8:73 65 7A 20 76 69 74 65   sez vit
11E0:00 43 68 61 69 73 65 73   .Chaise
11E8:20 2D 20 74 61 62 6C 65    - tabl
11F0:61 75 00 42 72 72 72 20   au.Brrr
11F8:21 21 21 00 46 9C 6C 69   !!!.F.l
1200:63 69 74 61 74 69 6F 6E   citatio
1208:73 20 21 00 42 4F 4F 4F   s !.BOO
1210:4F 55 4D 20 21 21 21 21   OUM !!!
1218:00 49 6C 20 66 61 69 74   .Il fai
1220:20 6E 6F 69 72 00 50 6F    noir.P
1228:72 74 65 20 6F 75 76 65   rte ouv
1230:72 74 65 00 43 6F 66 66   rte.Cof
1238:72 65 2D 66 6F 72 74 20   re-fort
1240:6F 75 76 65 72 74 00 43   ouvert.
1248:6F 72 64 65 20 70 65 6E   orde pe
1250:64 75 65 20 65 6E 20 57   due en
1258:00 2B 51 1B 3C 43 4C 45   .+Q.<CL
1260:46 81 54 4F 52 43 82 50   F.TORC.
1268:41 50 49 83 43 41 53 53   API.CAS
1270:84 43 4F 52 44 85 43 41   .CORD.C
1278:46 45 86 4D 41 52 54 87   FE.MART
1280:43 4C 45 53 88 42 55 52   CLES.BU
1288:49 89 4C 49 56 52 8A 54   I.LIVR.
1290:45 4E 41 8B 50 49 4C 45   ENA.PIL
1298:8C 50 4F 4D 4D 8D 53 45   :POMM.S
12A0:52 52 8E 50 4F 52 54 8F   RR.PORT
12A8:43 4F 46 46 90 43 41 56   COFF.CA
12B0:45 91 31 39 34 32 92 42   E.1942.
12B8:4F 4D 42 93 54 41 42 4C   OMB.TAB
12C0:94 4D 41 47 4E 95 43 4F   .MAGN.C
12C8:55 4C 96 43 4C 4F 55 97   UL.CLOU
12D0:50 4C 41 51 98 50 41 49   PLAQ.PA
12D8:4E 99 52 41 44 49 9A 54   N.RADI.
12E0:45 4C 45 9B 00 0F 13 76   ELE....
12E8:13 AB 13 CA 13 DF 13 EE   .+.J._.
```

FIGURE 37 : SUITE

```
12F0:13 14 14 4F 14 59 14 74
12F8:14 83 14 AD 14 B2 14 BC
1300:14 CA 14 DC 14 00 15 05
1308:15 52 15 65 15 99 15 FE
1310:0F 28 2B FE 0E 28 0B FE
1318:10 C2 69 15 11 08 17 C3
1320:E0 0A 3A 93 0C FE 17 C2
1328:DD 0A 3E 08 CD 50 38 20
1330:F6 3E 18 32 0F 16 3E 45
1338:32 4E 10 C3 E9 0A 3A 8C
1340:0C FE 01 20 08 3E 03 32
1348:8C 0C C3 E9 0A FE 0A 20
1350:04 3E 0E 18 F2 FE 0E 20
1358:10 3E 01 CD 50 18 28 05
1360:11 2A 17 18 BA 3E 0A 18
1368:DE FE 13 20 BA 3E 01 32
1370:92 0C 3E 0F 18 D1 FE 0E
1378:D2 DD 0A CD 48 18 3A 8C
1380:0C BE C2 DD 0A E5 21 9C
1388:0C 7E FE FF 28 08 23 FE
1390:80 20 F6 14 18 F3 E1 7A
1398:FE 05 38 09 11 1E 16 CD
13A0:3B 0B C3 27 0E 3E 80 77
13A8:C3 E9 0A FE 0E D2 DD 0A
13B0:CD 50 18 28 06 11 2F 16
13B8:C3 E0 0A 3A A7 0B FE 05
13C0:3A 8C 0C 20 E2 32 97 0C
13C8:18 DD FE 05 C2 E8 13 3A
13D0:97 0C FE 06 C2 DD 0A 3E
13D8:07 32 8C 0C C3 E9 0A FE
13E0:06 20 05 11 42 16 18 03
13E8:11 4C 16 C3 E0 0A FE 03
13F0:20 0B CD 50 18 C2 B5 13
13F8:11 5D 17 18 EE FE 0A 20
1400:0A CD 50 18 20 EF 11 63
1408:16 18 E0 FE 18 20 D9 11
1410:F3 17 18 D7 FE 14 20 18
1418:3A 94 0C FE 17 20 0C 32
1420:93 0C 32 A9 0C 11 CD 17
1428:C3 EC 0A 11 C0 17 18 BB
1430:FE 17 C2 69 15 3E 0B CD
1438:50 18 28 05 11 E0 17 18
1440:AA 3A 8C 0C FE 17 C2 DD
1448:0A 32 94 0C C3 E9 0A FE
1450:16 C2 E8 13 11 3E 17 18
1458:22 FE 14 20 05 11 7D 17
1460:18 19 FE 03 28 09 FE 0A
1468:28 05 11 6D 17 18 0C 11
1470:8E 17 18 07 FE 0F 20 06
```

FIGURE 37 : SUITE

```
1478:11 89 16 C3 E0 0A 11 7F   ...C‘..
1480:16 18 F8 FE 02 20 1C 3A   ..x~. .
1488:91 0C FE 02 20 05 11 B6   ..~. ..
1490:16 18 E8 FE 01 28 05 11   ..h~.(.
1498:A4 16 18 DF 3C 32 91 0C   $.._<2.
14A0:C3 E9 0A FE 1A C2 E8 13   Ci.~.Bh
14A8:11 1A 17 18 CE 11 72 16   ....N.r
14B0:18 C9 FE 0F C2 69 15 11   .I~.Bi.
14B8:AF 17 18 BF FE 19 20 05   /..?~.
14C0:11 03 18 18 B6 FE 0D C3   ....6~.
14C8:E1 13 FE 05 C2 E8 13 3A   a.~.Bh.
14D0:97 0C FE 06 C2 7E 14 32   ..~.B~.
14D8:8C 0C 18 52 FE 0F 28 05   ...R~.(
14E0:11 9B 17 18 6A 3A 8C 0C   ....j:.
14E8:FE 0F C2 E9 0A 3A 92 0C   ~.Bi.:.
14F0:FE 01 28 05 11 61 11 18   ~.(..a.
14F8:56 3E 13 32 8C 0C 18 2E   V>.2...
1500:11 9B 17 18 4A FE 13 20   ....J~.
1508:06 11 4E 17 C3 E0 0A FE   ..N.C‘.
1510:04 20 08 CD 50 18 C2 B5   .  .MP.B
1518:13 18 29 FE 1A 20 12 3A   ..)~. .
1520:95 0C FE 02 C2 97 14 DD   ..~.B..
1528:21 7C 18 CD 44 09 C3 E9   !|.MD.C
1530:0A FE 15 20 05 11 37 18   .~. ..7
1538:18 15 FE 1B C2 E8 13 11   ..~.Bh.
1540:27 18 18 0B 3A 8C 0C FE   '...:..
1548:08 C2 DD 0A 11 58 16 C3   .B]..X.
1550:E0 0A FE 02 C2 E8 13 3A   ‘.~.Bh.
1558:91 0C FE 02 C2 8E 14 3D   ..~.B..
1560:32 91 0C 18 C9 FE 0C 28   2...I~.
1568:05 11 C6 16 18 E1 3E 0C   ..F..a>
1570:CD 50 18 C2 B5 13 3A 8C   MP.B5.:
1578:0C FE 02 20 05 32 95 0C   .~. .2.
1580:18 11 3E 02 CD 50 18 28   ..>.MP.
1588:05 11 D5 16 18 C1 3E 01   ..U..A>
1590:32 91 0C AF 32 A7 0C 18   2../2'.
1598:95 FE 12 20 CC 3A 8C 0C   .~. L:.
15A0:FE 05 C2 DD 0A 3E 01 32   ~.B].>.
15A8:8F 0C 3E 05 32 A6 0C C3   ..>.2&.
15B0:E9 0A 40 40 40 00 00 00   i.@@@..
15B8:00 00 00 03 00 07 00 04   .......
15C0:02 08 00 05 03 00 00 00   .......
15C8:04 0A 00 00 00 0B 03 08   .......
15D0:00 0C 04 00 00 00 00 00   .......
15D8:00 00 06 00 00 0F 07 00   .......
15E0:00 00 00 0D 00 11 09 0D   .......
15E8:0D 00 00 00 00 00 0B 00   .......
15F0:00 14 10 11 00 15 0D 11   .......
15F8:10 16 00 00 00 00 00 14   .......
1600:12 14 10 15 13 15 11 00   .......
```

FIGURE 37 : SUITE

```
1608:14 00 12 17 00 00 13 00   .......
1610:00 00 00 19 17 00 00 00   .......
1618:00 00 00 00 00 FF 56 6F   ......
1620:75 73 20 70 6F 72 74 65   us port
1628:7A 20 74 72 6F 70 00 56   z trop.
1630:6F 75 73 20 6E 65 20 6C   ous ne
1638:27 61 76 65 7A 20 70 61   'avez p
1640:73 00 43 27 65 73 74 20   s.C'est
1648:62 6F 6E 00 51 75 65 6C   bon.Que
1650:6C 65 20 69 64 9C 65 00   le id.
1658:46 41 49 52 65 20 31 39   FAIRe
1660:34 32 00 43 27 65 73 74   42.C'e
1668:20 75 6E 20 70 6F 6C 61    un po
1670:72 00 47 72 61 6E 64 E5   r.Gran
1678:20 62 72 75 74 65 00 4F    brute
1680:68 20 68 69 73 73 65 2E   h hiss
1688:00 45 73 73 61 79 65 72   .Essay
1690:20 4F 55 56 52 65 00 43    OUVRe
1698:27 65 73 74 20 62 6C 6F   'est b
16A0:71 75 9C 00 49 6C 20 66   qu..Il
16A8:61 75 74 20 64 65 73 20   aut de
16B0:70 69 6C 65 73 00 43 27   piles.
16B8:65 73 74 20 64 9C 6A 61   est d.
16C0:20 66 61 69 74 00 4A 65    fait.
16C8:20 6E 65 20 73 61 69 73    ne sa
16D0:20 70 61 73 00 4F 75 20    pas.O
16D8:63 65 6C 61 20 3F 00 0A   cela ?
16E0:20 42 72 61 76 6F 2E 2E    Bravo
16E8:2E 20 41 20 62 69 65 6E   . A bi
16F0:74 6F 74 00 0A 20 50 61   tot..
16F8:73 20 64 65 20 63 68 61   s de c
1700:6E 63 65 2E 2E 2E 20 00   nce...
1708:46 41 49 52 65 20 63 6F   FAIRe
1710:6D 62 69 6E 61 69 73 6F   mbinai
1718:6E 00 45 73 73 61 79 65   n.Essa
1720:7A 20 45 43 4F 55 74 65   z ECOU
1728:72 00 41 76 6F 69 72 20   r.Avoi
1730:6C 61 20 62 6F 6E 6E 65   la bon
1738:20 63 6C 65 46 00 54 72    cleF.
1740:6F 70 20 64 61 6E 67 65   op dan
1748:75 72 65 75 78 D8 54 69   ureuxX
1750:63 20 54 61 63 20 54 69   c Tac
1758:63 2E 2E 2E 00 45 4E 4C   c....E
1760:45 76 65 72 20 54 41 42   Ever T
1768:4C 65 61 75 00 54 6F 75   Leau.T
1770:74 20 65 73 74 20 6E 6F   t est
1778:72 6D 61 6C 00 43 27 65   rmal.C
1780:73 74 20 75 6E 65 20 63   st une
1788:72 6F 75 74 65 00 45 73   route.
```

FIGURE 37 : SUITE

```
1790:73 61 79 65 7A 20 4C 49   sayez L
1798:52 45 00 55 74 69 6C 69   RE.Util
17A0:73 65 7A 20 4E 2C 53 2C   sez N,S
17A8:45 20 6F 75 20 57 00 45   E ou W.
17B0:6C 6C 65 20 65 73 74 20   lle est
17B8:63 6F 69 6E 63 9C 65 00   coinc.e
17C0:49 6C 20 65 73 74 20 63   Il est
17C8:6C 6F 75 9C 00 49 6C 20   lou..Il
17D0:79 20 61 20 75 6E 65 20   y a une
17D8:73 65 72 72 75 72 65 00   serrure
17E0:49 6C 20 66 61 75 74 20   Il faut
17E8:64 65 73 20 6F 75 74 69   des out
17F0:6C 73 00 45 6E 74 72 9C   ls.Entr
17F8:65 20 67 72 61 74 75 69   e gratu
1800:74 65 00 49 6C 20 65 73   te.Il e
1808:74 20 72 61 73 73 69 73   t rassi
1810:00 0A 56 6F 75 73 20 65   ..Vous
1818:6E 74 72 65 7A 20 64 61   ntrez d
1820:6E 73 20 2E 2E 2E 00 50   ns ....
1828:61 73 20 64 65 20 74 6F   as de t
1830:6E 61 6C 69 74 FC 00 45   nalit!.
1838:43 4F 55 74 65 72 20 43   COUter
1840:41 53 53 65 74 74 65 00   ASSette
1848:21 9B 0C 5F 16 00 19 C9   !.._...
1850:CD 48 18 3E 80 BE C9 CD   MH.>.>I
1858:AF 09 21 DF 16 CD F7 FE   /.!_.Mw
1860:DD 21 AB 18 CD 44 09 CD   ]!+.MD.
1868:2D 09 C3 EC 09 CD AF 09   -.Cl.M/
1870:21 F4 16 CD F7 FE DD 21   !t.Mw~]
1878:CE 18 18 E8 11 04 11 04   N..h...
1880:11 04 0D 10 00 10 0F 04   .......
1888:0F 04 0F 04 0C 10 00 10   .......
1890:11 04 11 04 11 04 11 04   .......
1898:0D 04 12 04 12 04 12 04   .......
18A0:11 04 19 04 19 04 19 04   .......
18A8:16 10 FF 14 0C 14 04 16   .......
18B0:08 14 08 00 08 14 0C 14   .......
18B8:04 16 08 14 08 00 08 16   .......
18C0:08 14 08 12 08 11 08 0F   .......
18C8:10 0D 0C 0D 04 FF 0D 10   .......
18D0:0D 0C 0D 04 0D 0C 10 04   .......
18D8:10 0C 0F 04 0F 0C 0D 04   .......
18E0:0D 08 0C 08 0D 10 FF 31   .......
18E8:33 00 00 00 00 00 00 00   3......
```

FIGURE 37 : FIN

```
5 REM *** CHARGEUR - VIDEO ***
10 CLEAR50,&H7F0:DEFINT A-Z
20 INIT#1,"CASI:"
30 INPUT#1,N$,D,F
40 MOTOR
50 PRINT"Trouv :";N$
60 FORI=D-1TOF
70 POKEI,INP(#1)
80 NEXT:MOTOR
90 END
100 CLEAR50,&H7F0:DEFINT A-Z
110 D=&H800:F=&H18E8
120 N$="AVENT":INIT#1,"CASO:"
130 INPUT"Magnto OK";T$
140 PRINT#1,N$,D,F:MOTOR
150 FORI=1TO1800:NEXT
160 FORI=DTOF:OUT#1,PEEK(I)
170 NEXT:MOTOR
180 END
```

```
5 REM *** ENTREUR DE CODES - VIDEO
10 CLEAR50,&H7FF:A=&H800
20 PRINTHEX$(A);" : ";:INPUTC$
30 V=VAL("&H"+C$):POKEA,V
49 A=A+1:IF A>&H18E8 THEN PRINT"TER
...":BEEP2,3:END
50 GOTO20
```

FIGURE 38 : CHARGEURS

OTHELLO-REVERSI

Remarquable à la fois par la simplicité de ses règles et par la complexité des tactiques et des stratégies à mettre en oeuvre , **Othello-Reversi** a conquis en quelques années des dizaines de millions de passionnés dans le monde
Bien qu'il reste relativement peu pratiqué en France , il possède toutes les qualités d'un jeu exemplaire .

Le Reversi a été inventé dans la seconde moitié du 19ème siècle en Angleterre . Après de nombreuses aventures , un japonais l'a "réinventé" sous le nom d'Othello ...

Les règles du jeu sont simples . Othello se joue sur **un plateau 8*8** dont toutes les cases sont de la même couleur . Les pions utilisés sont à double face : d'un coté blanc , de l'autre coté noir . Chaque joueur prend une couleur : l'un sera blanc , l'autre noir . Au début de la partie , les joueurs disposent quatre pions au centre de l'échiquier .

Les premiers pions posés , les joueurs jouent chacun leur tour en posant un pion de leur couleur sur une case libre , de telle façon qu'entre ce pion posé et un autre pion de la même couleur , un pion au moins de la couleur adverse soit pris en "tenaille" , selon une ligne horizontale , verticale ou diagonale . Le ou les pions pris sont alors retournés et ils prennent la couleur du preneur , d'où le nom de Reversi . La pose d'un pion peut prendre en tenaille des pions selon plusieurs lignes . Ils sont alors tous retournés . Mais il n'y a pas "transitivité" : un pion retourné ne peut entraîner de nouveaux retournements.

Il y a obligation de toujours retourner un pion adverse . Lorsqu'un joueur ne peut pas jouer de coup qui le lui permette , il doit passer son tour . Lorsqu'aucun joueur ne peut plus jouer , soit que toutes les cases soient remplies , soit qu'il n'y ait plus de coup possible selon la rêgle ci-dessus , la partie s'arrête et le vainqueur est celui dont la couleur apparait sur le plus grand nombre de pions posés .

Sur le CANON X-07 , le logiciel suit scrupuleusement les règles décrites ci-dessus . Les pions ne pouvant être colorés , vous posséderez des pions de forme différente au programme qui se révèle , soit dit en passant , très fort ...

Le mode d'emploi est le suivant :

_ Après avoir chargé le programme en mémoire grâce à la K7 (CLOAD puis RUN pour charger les codes) ou à la "main" grâce aux chargeurs joints , vous devez lancer le programme par **EXEC &HE00** suivi de RETURN .

_ Une présentation défile ... Tapez sur RETURN : le X-07 vous demande si vous désirez jouer contre lui ou contre un adversaire humain .

_ Ensuite , le niveau de jeu vous est demandé : entre **1 et 9** . Le niveau 1 est beaucoup plus facile que le 9 ...

_ Le X-07 s'enquiert enfin de vos initiales (4 lettres) : tapez-les et le CANON vous présentera le damier .

_ Sur la gauche de l'écran , l'échiquier 8*8 est dessiné et à droite de l'afficheur le nombre de pions gagnés par chacun des deux joueurs .

_ Pour jouer , vous devez déplacer le point clignotant avec les quatre curseurs ... Dès que vous avez choisi votre place , tapez sur RETURN et le X-07 calculera et affichera les pions gagnés . Le X-07 jouera ensuite et le jeu continuera ainsi jusqu'à la fin de la partie .

_ Notons que si vous devez passer , vous pouvez appuyer sur **"P"** : le X-07 vous accordera ce droit si vous êtes vraiment bloqué ... Sinon , il vous affichera vos possibilités de jeu ! De plus , le jeu peut être quitté à tout moment par l'habituel **"CTRL Q"** .

Bonne chance ... Dommage , on ne peut pas tricher !!

Longueur de REVERSI : 3073 octets .
Implantation de REVERSI : de 800h à 1400h .

```
0800:4D 4B 4B 4B 4B 4B 4B 54   MKKKKKKT
0808:50 5E 48 48 48 48 48 48   P^HHHHHH
0810:52 50 5E 48 48 48 48 48   RP^HHHHH
0818:48 52 50 5E 48 48 48 48   HRP^HHHH
0820:48 48 52 50 5E 48 48 48   HHRP^HHH
0828:48 48 48 52 50 5E 48 48   HHHRP^HH
0830:48 48 48 48 52 50 5E 48   HHHHRP^H
0838:48 48 48 48 48 52 50 60   HHHHHRP`
0840:58 58 58 58 58 58 5A 50   XXXXXXZP
0848:F6 F7 F8 FF 08 09 0A 01   vwx.....
0850:00 00 F7 F6 FF 08 09 00   ..wv....
0858:01 F8 F7 F6 FF 00 09 0A   .xwv....
0860:01 F8 F7 00 16 FD 04 02   .xw..}..
0868:02 04 FD 16 F0 FD F6 FF   ..}.p}v.
0870:FF FF FF F6 FD F0 04 FF   ...v}p..
0878:00 00 00 00 FF 04 F0 02   ......p.
0880:FF 00 00 00 00 FF 02 F0   .......p
0888:02 FF 00 00 00 00 FF 02   ........
0890:F0 04 FF 00 00 00 00 FF   p.......
0898:04 F0 FD F6 FF FF FF FF   .p}v....
08A0:F6 FD F0 16 FD 04 02 02   v}p.}...
08A8:04 FD 16 16 FD 04 02 02   .}..}...
08B0:04 FD 16 F0 FD F6 00 00   .}.p}v..
08B8:00 00 F6 FD F0 04 00 00   ..v}p...
08C0:00 00 00 00 04 F0 02 00   .....p..
08C8:00 00 00 00 00 02 F0 02   ......p.
08D0:00 00 00 00 00 00 02 F0   .......p
08D8:04 00 00 00 00 00 00 04   ........
08E0:F0 FD F6 00 00 00 00 F6   p}v....v
08E8:FD F0 16 FD 04 02 02 04   }p.}....
08F0:FD 16 01 01 01 01 01 01   }.......
08F8:01 01 F0 01 01 01 01 01   ..p.....
0900:01 01 01 F0 01 01 01 01   ...p....
0908:01 01 01 01 F0 01 01 01   ....p...
0910:01 01 01 01 01 F0 01 01   .....p..
0918:01 01 01 01 01 01 F0 01   ......p.
0920:01 01 01 01 01 01 01 F0   .......p
0928:01 01 01 01 01 01 01 01   ........
0930:F0 01 01 01 01 01 01 01   p.......
0938:01 00 FD 7E 00 32 80 09   ..}~.2..
0940:32 9C 09 32 C3 09 C9 DD   2..2C.I]
0948:7E 02 CD 3D 09 CD 75 12   ~.M=.Mu.
0950:3E 16 32 CF 09 AF 32 CE   >.2O./2N
0958:09 B6 28 02 AF C9 4D 16   .6(./IM.
0960:08 5D 1A 5F 1A 57 06 FF   .]._.W..
0968:3A CE 09 C6 0C 32 CE 09   :N.F.2N.
0970:AF 32 D0 09 32 D1 09 7D   /2P.2Q.}
0978:82 6F 7E B7 28 2A 04 E6   .o~7(*.f
0980:01 28 F4 78 B7 28 21 3A   .(tx7(!:
0988:D0 09 80 32 D0 09 7D 92   P..2P.}.
```

FIGURE 39 : OTHELLO

```
0990:6F E5 2A CE 09 77 23 22   oe*N.w#"
0998:CE 09 E1 36 01 25 3A D1   N.a6.%:Q
09A0:09 86 32 D1 09 24 10 E6   ..2Q.$.f
09A8:06 FF 69 1C 16 08 1A 57   ..i....W
09B0:B7 20 C4 3A D0 09 B7 C8   7 D:P.7H
09B8:25 3A D1 09 87 86 32 D1   %:Q...2Q
09C0:09 24 36 01 7D F6 80 E5   .$6.}v.e
09C8:2A CE 09 77 E1 C9 00 16   *N.waI..
09D0:00 00 3A 01 13 4F DD 46   ..:..O]F
09D8:03 78 FE 03 38 18 C5 06   .x~.8.E.
09E0:02 CD F6 09 C1 79 80 D6   .Mv.Ay.V
09E8:32 38 0B 1F 80 47 81 D6   28...G.V
09F0:43 38 03 2F 80 47 C5 78   C8./.GEx
09F8:81 21 64 08 FE 24 38 03   .!d.~$8.
0A00:21 AB 08 FE 3C 38 03 21   !+.~<8.!
0A08:F2 08 11 80 15 01 47 00   r.....G.
0A10:ED B0 C1 FD 21 80 15 DD   m0A}!..]
0A18:4E 02 11 80 00 21 80 15   N....!..
0A20:FD 19 19 FD 71 00 79 EE   }..}q.yn
0A28:81 4F 05 FD 70 01 FD 74   .O.}p.}t
0A30:02 FD 75 03 20 EA 3A 01   .}u. j:.
0A38:16 F6 80 32 01 16 DD E5   .v.2..]e
0A40:DD 21 EA 13 21 52 13 16   ]!j.!R..
0A48:00 ED 5F DD A6 00 5F 7E   .m_]&._~
0A50:19 46 77 ED 52 70 DD 5E   .FwmRp]^
0A58:00 1C 19 DD 23 DD 7E 00   ...]#]~.
0A60:B7 20 E6 26 15 DD 21 04   7 f&.]!.
0A68:13 DD 7E 00 DD 23 6F 17   .]~.]#o.
0A70:38 0C 7E B7 28 F3 CB FD   8.~7(sK}
0A78:36 01 CB BD 18 EB FD 21   6.K=.k}!
0A80:11 13 DD 21 52 13 DD 7E   ..]!R.]~
0A88:00 DD 23 FD 77 00 6F 17   .]#}w.o.
0A90:38 08 7E B7 20 F0 FD 23   8.~7 p}#
0A98:18 EC FD 21 80 15 3E FF   .l}!..>.
0AA0:47 CD BE 0A DD E1 C9 FD   GM>.]aI}
0AA8:7E 06 ED 44 FD BE 09 38   ~.mD}>.8
0AB0:03 FD 7E 09 FD 46 08 FD   .}~.}F.}
0AB8:CB 01 7E C4 5D 12 11 80   K.~D]...
0AC0:00 FD 19 FD 77 08 FD 70   .}.}w.}p
0AC8:09 DD 21 10 13 CD 3A 09   .]!..M:.
0AD0:FD 36 06 01 FD 7E 01 E6   }6..}~.f
0AD8:0F 20 06 CD A3 0B FD 77   . .M#.}w
0AE0:07 FD 7E 04 FD 77 0A 6F   .}~.}w.o
0AE8:18 0E DD 23 DD 7E 00 FD   ..]#]~.}
0AF0:BE 0A 28 F6 6F 17 38 59   >.(vo.8Y
0AF8:AF B6 20 EE E5 FD 6E 03   /6 ne}n.
0B00:FD 66 02 22 CE 09 E1 CD   }f."N.aM
0B08:5E 09 28 DE FD 7E 01 E6   ^.(^}~.f
0B10:0F 20 08 3A D1 09 FD 86   . .:Q.}.
0B18:07 18 0F DD E5 FD E5 E5   ...]e}ee
```

FIGURE 39 : SUITE

```
0B20:CD A7 0A E1 FD E1 DD E1   M'.a}a]
0B28:2F 3C F5 CD 3A 09 EE 81   /<uM:.n
0B30:32 9A 0B FD 56 02 FD 5E   2..}V.}
0B38:03 CD 8E 0B F1 FD BE 06   .M..q}>
0B40:38 A8 28 A6 FD 77 06 FD   8((&}w.
0B48:75 04 FD BE 08 38 9B 37   u.}>.8.
0B50:C9 FD 7E 06 FD 6E 04 3D   I}~.}n.
0B58:28 02 3C C9 FD CB 01 7E   (.<I}K.
0B60:C0 FD 7E 00 CD 83 0B 3E   @}~.M..
0B68:02 C0 FD 7E 00 EE 81 CD   .@}~.n.
0B70:83 0B 3E FE C0 AF FD B6   . >~@/}
0B78:01 FD 7E 07 C8 CD A7 0A   .}~.HM'
0B80:ED 44 C9 2E 00 06 47 BE   mDI.. G
0B88:C8 23 10 FB B7 C9 3E 0C   H#.{7I>
0B90:83 3F 1A 13 FE 80 30 05   ._ ~.0
0B98:6F 36 01 18 F5 CB BF 6F   o6..uK?
0BA0:36 00 C9 2E 00 06 47 0E   6.I...G
0BA8:80 11 00 14 7E B7 28 03   ....~7(
0BB0:1F 1A 38 02 2F 3C 81 4F   ..8./<.
0BB8:2C 1C 10 F0 79 FD CB 00   ...py}K
0BC0:46 C0 2F 3C C9 AF CD 0A   F@/<I/M
0BC8:C3 28 FA C9 E5 D5 C5 FD   C(zIeUE
0BD0:E5 21 DF 0B E5 FD 21 20   e!_.e}!
0BD8:00 21 00 02 CD DF C2 FD   .!..M_B
0BE0:E1 C1 D1 E1 C9 11 3B 0D   aAQaI.;
0BE8:21 04 0A D5 0E 26 CD 31   !..U. M
0BF0:C2 D1 1A B7 C8 13 CD 8A   BQ.7H.M
0BF8:C1 18 F7 E5 D5 C5 32 FF   A.weUE2
0C00:0D 21 C8 01 77 3E 28 01   .!H.w>(
0C08:01 01 11 C6 01 CD 2F C9   ...F.M/
0C10:1A B7 18 CD C5 3E 14 01   .7.ME>.
0C18:00 04 CD 2F C9 C1 10 F4   ..M/IA.
0C20:C9 06 0C 21 33 06 C5 3E   I..!3.E
0C28:1A 01 00 09 CD 2F C9 C1   ....M/I
0C30:10 F4 C9 E0 FC 80 80 80   .tI`|.
0C38:FC 80 80 80 E1 FC 80 80   |...a|.
0C40:80 FC A8 90 A8 E2 FC 80   .|(.(b|
0C48:80 80 FC 90 B8 90 E3 FC   ..|.8.c
0C50:90 B8 90 FC 80 80 80 E4   .8.|...
0C58:FC 90 B8 90 FC A8 90 A8   |.8.|(.
0C60:E5 FC 90 B8 90 FC 90 B8   e|.8.|.
0C68:90 E6 FC A8 90 A8 FC 80   .f|(.(|
0C70:80 80 E7 FC A8 90 A8 FC   ..g|(.(
0C78:A8 90 A8 E8 FC A8 90 A8   (.(h|(.
0C80:FC 90 B8 90 E9 00 00 7C   |.8.i..
0C88:44 54 44 7C 00 EA 00 00   DTD|.j.
0C90:50 20 50 00 00 00 EB 00   P P...k
0C98:00 20 70 20 00 00 00 CD   . p ...
0CA0:E8 0B 01 00 00 0B 78 B1   h....x
0CA8:20 FB C3 D9 08 D9 F1 FE   {CY.Yq
```

```
0CB0:11 C2 9B C7 08 D9 ED 57
0CB8:E2 C3 C3 CD BD C0 3A 2B
0CC0:00 06 08 B0 32 2B 00 3E
0CC8:0C D3 BB CD A2 00 C3 CF
0CD0:C3 E5 D5 C5 22 50 04 01
0CD8:07 07 21 C4 01 CD 5F BE
0CE0:FD E5 E1 11 C7 01 1A FE
0CE8:30 20 03 3E 20 12 CD EB
0CF0:0B C1 D1 E1 C9 06 04 21
0CF8:02 01 C5 CD C5 0B 12 13
0D00:4F D5 CD 31 C2 D1 24 C1
0D08:10 F0 C9 30 00 30 1F 57
0D10:00 57 10 00 02 77 02 00
0D18:1D 77 1D 02 00 02 1F 75
0D20:00 75 1F 52 20 49 20 58
0D28:00 52 65 76 65 72 73 69
0D30:20 70 6F 75 72 20 58 2D
0D38:30 37 00 20 20 20 20 20
0D40:20 20 20 20 20 00 20 2A
0D48:20 4A 6F 75 65 7A 2D 76
0D50:6F 75 73 20 63 6F 6E 74
0D58:72 65 52 69 78 20 28 4F
0D60:2F 4E 29 00 49 6E 69 74
0D68:69 61 6C 65 73 20 6A 6F
0D70:75 65 75 72 20 6E DF 32
0D78:23 23 23 23 00 52 69 78
0D80:20 52 69 78 20 00 4E 69
0D88:76 65 61 75 20 64 65 20
0D90:6A 65 75 20 28 31 2D 39
0D98:29 00 45 6E 74 72 65 7A
0DA0:20 76 6F 73 20 69 6E 69
0DA8:74 69 61 6C 65 73 23 23
0DB0:23 23 00 41 4E 44 52 00
0DB8:47 41 47 4E 45 00 45 47
0DC0:41 4C 49 54 45 20 21 00
0DC8:4F 6E 20 72 65 63 6F 6D
0DD0:6D 65 6E 63 65 20 28 4F
0DD8:2F 4E 29 00 50 61 73 73
0DE0:65 00 49 6E 74 65 72 64
0DE8:69 74 00 50 65 72 6D 69
0DF0:73 20 E9 00 41 72 72 65
0DF8:74 20 4F 2F 4E 3F 00 50
0E00:3E 33 CD 28 E4 3E 38 CD
0E08:28 E4 F3 21 AB 0C 22 3D
0E10:00 FB CD BD C0 CD 9E CE
0E18:CD 21 0C CD CC 0B 06 04
0E20:21 13 0D CD 14 0C 21 02
0E28:08 11 23 0D CD EB 0B 21
0E30:03 02 11 29 0D CD EB 0B
0E38:1E 13 0E 27 06 07 21 C4
```

FIGURE 39 : SUITE

FIGURE 39 : SUITE

```
0E40:01 71 23 70 2B CD 23 10  .q#p+M#.
0E48:01 78 FE 10 20 F0 0C 29  .x~. p.)
0E50:FE 4A 20 E8 CD C5 0B CD  ~J hME.M
0E58:CC 0B 31 C0 01 CD 9E CE  L.1@.M.N
0E60:3E 2A 32 9F 13 3E 20 32  >*2..> 2
0E68:98 13 26 16 2E 04 11 80  ..&.....
0E70:00 2E F6 06 12 36 00 2C  ..v..6.,
0E78:71 2D 0C 20 01 0D 13 10  q-. ....
0E80:F4 21 46 0D CD F7 FE AF  t!F.Mw~/
0E88:32 97 13 32 9E 13 CD C5  2..2..ME
0E90:0B FE 4E 20 11 CD 9E CE  .~N .M.N
0E98:21 64 0D CD F7 FE 11 81  !d.Mw~..
0EA0:0D CD F5 0C 18 26 32 9E  .Mu..&2.
0EA8:13 CD 9E CE 21 7D 0D 11  .M.N!}..
0EB0:B1 0D 01 04 00 ED B0 21  1.....m0!
0EB8:8C 0D CD F7 FE CD C5 0B  ..Mw~ME.
0EC0:FE 3A 30 F9 D6 31 38 F5  ~:0yV18u
0EC8:3C 32 9D 13 CD 9E CE 21  <2..M.N!
0ED0:9A 0D CD F7 FE 11 B3 0D  ..Mw~.3.
0ED8:CD F5 0C CD 9E CE CD FC  Mu.M.NM|
0EE0:11 21 9A 13 22 78 10 21  .!.."x.!
0EE8:1D 15 22 FD 12 CD 14 12  .."}.M..
0EF0:06 02 21 0B 0D CD 14 0C  ..!..M..
0EF8:21 01 0B 11 81 0D CD EB  !.....Mk
0F00:0B 21 01 10 11 B3 0D CD  .!...3.M
0F08:EB 0B 3A 9E 13 B7 28 0C  k.:..7(.
0F10:3A 9D 13 C6 30 4F 21 01  :..F0O!.
0F18:0E CD 31 C2 0E EA 21 02  .M1B.j!.
0F20:0B CD 31 C2 0E EB 21 02  .M1B.k!.
0F28:10 CD 31 C2 CD 38 0F 21  .M1BM8.!
0F30:0F 10 22 B8 00 C3 70 10  .."8.Cp.
0F38:21 00 15 11 00 00 06 47  !......G
0F40:7E CB 47 28 01 14 CB 7F  ~KG(..K.
0F48:28 01 1C 2C 10 F2 21 FF  (..,.r!.
0F50:12 73 23 72 7A 83 2C 77  .s#rz.,w
0F58:26 00 6B FD 21 02 0D CD  &.k}!..M
0F60:D1 0C 6A FD 21 02 12 CD  Q.j}!..M
0F68:D1 0C 7A B7 28 17 7B B7  Q.z7(.{7
0F70:28 13 82 FE 40 28 0E C9  (..~@(.I
0F78:3A 03 13 B7 CA 6A 10 DD  :..7Jj.]
0F80:BE 06 CA 6A 10 3A FF 12  >.Jj.:..
0F88:47 3A 00 13 90 11 B3 0D  G:....3.
0F90:28 42 30 03 11 81 0D CD  (B0....M
0F98:E8 0B 21 04 0F 11 B8 0D  h.!...8.
0FA0:CD EB 0B CD A2 0C CD C5  Mk.M".ME
0FA8:0B CD 9E CE 21 C8 0D CD  .M.N!H.M
0FB0:F7 FE CD C5 0B FE 4F CA  w~ME.~OJ
0FB8:5A 0E 3E 32 CD 28 E4 3E  Z.>2M(d>
0FC0:39 CD 28 E4 3E 1C CD 28  9M(d>.M(
0FC8:E4 F3 21 99 C7 22 3D 00  ds!.G"=.
```

FIGURE 39 : SUITE

```
0FD0:FB C3 26 C5 11 BE 0D CD
0FD8:E8 0B 18 C7 CD 43 10 0D
0FE0:18 10 CD 43 10 04 18 0A
0FE8:CD 43 10 05 18 04 CD 43
0FF0:10 0C 28 E6 07 47 29 E6
0FF8:07 4F C5 C6 F7 04 C6 09
1000:10 FC 6F 22 FD 12 C1 CD
1008:10 10 CD 2D 10 C3 AE 10
1010:78 87 87 C6 02 32 B9 00
1018:29 87 87 81 81 C6 03 32
1020:B3 00 C9 C5 7B 01 00 02
1028:CD 2F C9 C1 C9 E5 21 B8
1030:00 3E 13 C5 01 00 02 CD
1038:2F C9 01 00 30 CD A5 0C
1040:C1 E1 C9 CD 54 10 06 FF
1048:7D 04 D6 09 30 FB C6 09
1050:4F E6 07 C9 7E B7 28 04
1058:3E 11 18 02 3E 12 E5 01
1060:00 02 21 B8 00 CD 2F C9
1068:E1 C9 DD 7E 06 32 03 13
1070:31 C0 01 CD CC 0B DD 2A
1078:93 13 DD 22 78 10 3A 9F
1080:13 32 98 13 EE 0A 32 9F
1088:13 21 01 0A 4F CD 31 C2
1090:3A 98 13 21 01 14 4F CD
1098:31 C2 2A FD 12 CD 46 10
10A0:CD 10 10 DD 7E 04 B7 28
10A8:05 CD 3A 11 18 C2 31 C0
10B0:01 2A FD 12 CD 2D 10 E5
10B8:CD 3D C0 3E 82 11 C4 01
10C0:01 01 20 CD 2F C9 1A E1
10C8:FE 33 CA EE 0F FE 37 CA
10D0:DC 0F FE 35 CA E2 0F FE
10D8:31 CA E8 0F 3E 2D CD FB
10E0:0B 20 11 FD 21 80 14 CD
10E8:18 12 3E 2D CD FB 0B 28
10F0:F9 CD 14 12 3E 0D CD FB
10F8:0B 28 08 3E 20 CD FB 0B
1100:C2 D7 11 CD 47 09 20 0F
1108:E5 11 E2 0D CD 9F 0C CD
1110:E5 0B 3C E1 C3 92 12 CD
1118:54 11 3A FF 0D FE 0D CA
1120:70 10 3E 20 CD FB 0B 28
1128:F9 21 80 14 11 00 15 01
1130:47 00 ED B0 CD B9 11 C3
1138:AE 10 CD D2 09 26 15 CD
1140:47 09 C2 54 11 11 DC 0D
1148:CD 9F 0C CD 14 12 CD E5
1150:0B C3 78 0F 22 FD 12 CD
1158:46 10 AF 11 80 00 ED 52
```

FIGURE 39 : SUITE

```
1160:32 03 13 11 09 00 0C CB   2.......
1168:28 C5 38 04 19 47 18 05   (E8..G.
1170:AF ED 52 06 06 3A 9F 13   /mR..:.
1178:FE 20 28 04 3E 0C 80 47   ~ (.>..
1180:0E 00 7E B7 28 08 0E 02   ..~7(..
1188:FE 01 28 02 0E 04 78 81   ~.(...x
1190:5F FD 21 BF 11 FD 19 C1   _}!?.}.
1198:04 61 68 06 03 C5 FD 4E   .ah..E}
11A0:01 CD 31 C2 01 00 40 CD   .M1B..@
11A8:A5 0C FD 4E 00 CD 31 C2   %.}N.M1
11B0:01 00 40 CD A5 0C C1 10   ..@M%.A
11B8:E4 CD 14 12 C3 38 0F E0   dM..C8.
11C0:E3 E2 E5 E1 E4 E0 E2 E3   cbead'b
11C8:E5 E6 E8 E0 E6 E2 E8 E1   efh'fbh
11D0:E7 E0 E1 E3 E4 E6 E7 3E   g'acdfg
11D8:41 CD FB 0B 20 11 11 F4   AM{. ..
11E0:0D CD 9F 0C CD C5 0B FE   .M..ME.
11E8:4F CA BA 0F CD E5 0B 3E   OJ:.Me.
11F0:50 CD FB 0B 20 03 C3 92   PM{. .C
11F8:12 C3 AE 10 21 00 15 36   .C..!..
1200:00 2C 20 FB 3E 01 32 1F   ., {>.2
1208:15 32 27 15 0F 32 1E 15   .2'..2.
1210:32 28 15 C9 FD 21 00 15   2(.I}!.
1218:E5 D5 C5 2E 01 16 04 26   eUE....
1220:01 1E 08 D5 CD 3E 12 CD   ...UM>.
1228:31 C2 D1 24 FD 23 1D 20   1BQ$}#.
1230:F2 2C 01 0A 00 FD 09 15   r,...}.
1238:20 E5 C1 D1 E1 C9 FD 7E    eAQaI}
1240:00 B7 28 08 FE 01 28 08   .7(.~.(
1248:0E E6 18 06 0E E0 18 02   .f...'.
1250:0E E3 FD 7E 09 B7 C8 0C   .c}~.7H
1258:FE 80 C8 0C C9 E5 D5 C5   ~.H.IeU
1260:F5 21 80 15 11 00 14 01   u!.....
1268:47 00 ED B0 F1 18 CB D5   G.m0q.K
1270:11 A1 13 18 04 D5 11 80   .!...U.
1278:14 E5 C5 21 00 15 01 47   .eE!...
1280:00 ED B0 C1 E1 D1 C9 D5   .m0AaQI
1288:E5 C5 21 A1 13 11 00 15   eE!!...
1290:18 EC FE 01 28 08 3A 01   .l~.(.:
1298:13 FE 04 CA 70 10 CD 14   .~.Jp.M
12A0:12 CD 6F 12 FD 21 52 13   .Mo.}!R
12A8:2A B8 00 E5 26 15 DD 7E   *8.e&..
12B0:02 0E 00 CD 3D 09 FD 7E   ...M=.}
12B8:00 FD 23 6F 17 38 1A C5   .}#o.8.
12C0:E5 CD 50 09 E1 C1 28 EE   eMP.aA(
12C8:0C CD 87 12 C5 CD 46 10   .M..EMF
12D0:CD 10 10 CD 58 10 C1 18   M..MX.A
12D8:DD 79 B7 E1 22 B8 00 F5   ]y7a"8.
12E0:28 05 11 EB 0D 18 03 11   (..k...
12E8:DC 0D CD E8 0B CD C5 0B   \.Mh.ME
```

FIGURE 39 : SUITE

```
12F0:CD E5 0B CD 14 12 F1 C2
12F8:AE 10 C3 78 0F 17 15 06
1300:04 0A 01 00 00 07 3F 46
1308:01 09 06 10 40 36 45 3D
1310:FF 00 07 46 3F 12 44 02
1318:41 19 34 05 2D 17 32 2F
1320:43 03 04 42 2B 24 22 1B
1328:26 29 20 31 30 3A 25 0C
1330:1C 2A 21 0D 39 33 0B 18
1338:13 2E 3B 38 10 01 06 09
1340:3D 45 36 40 0F 0A 3C 37
1348:FF 37 FF 0F FF 33 09 4C
1350:44 09 00 07 46 3F 12 44
1358:02 41 19 34 05 2D 17 14
1360:32 2F 43 03 04 42 2B 24
1368:22 1B 16 15 26 29 1D 20
1370:31 30 3A 25 0C 1C 2A 21
1378:0D 39 33 0B 18 13 0E 2E
1380:3B 38 10 01 06 09 3D 45
1388:36 40 0F 0A 3C 37 1F 1E
1390:28 27 FF 9A 13 01 02 00
1398:2A 01 93 13 80 01 4F 20
13A0:02 00 00 00 00 00 00 00
13A8:00 00 00 00 00 00 00 01
13B0:00 00 00 00 00 80 80 80
13B8:80 00 00 00 00 00 01 01
13C0:80 00 00 00 00 00 00 00
13C8:01 80 00 00 00 00 00 00
13D0:00 00 00 00 00 00 00 00
13D8:00 00 00 00 00 00 00 00
13E0:00 00 00 00 00 00 00 00
13E8:4C 44 03 07 03 07 07 07
13F0:07 07 03 00 05 0A 43 50
13F8:35 09 4C 44 09 41 2C 4C
1400:16 00 00 00 00 00 00 00
```

FIGURE 39 : FIN

```
5 REM *** OTHELLO - REVERSI ***
10 CLEAR50,&H7FF
20 INIT#1,"CASI:"
30 INPUT#1,N$,D,F
40 MOTOR
50 PRINT"Trouv :";N$
60 FORI=D-1TOF
70 POKEI,INP(#1)
80 NEXT:MOTOR
90 END
100 CLEAR50,&H7FF
110 D=&H800:F=&H1400
120 N$="AVENT":INIT#1,"CASO:"
130 INPUT"Magnto OK";T$
140 PRINT#1,N$,D,F:MOTOR
150 FORI=1TO1800:NEXT
160 FORI=DTOF:OUT#1,PEEK(I)
170 NEXT:MOTOR
180 END
```

```
5 REM *** CHARGEUR DE CODES

10 CLEAR,&H7FF:A=&H800
20 PRINTHEX$(A);" : ";:INPUT C$
30 V=VAL("&H"+C$):POKEA,V
49 A=A+1:IFA>&H1400 THEN PRINT" -
.. ":BEEP2,3:END
50 GOTO20
```

FIGURE 40 : CHARGEURS

CONCLUSION

En l'espace de six mois , nous avons concocté deux ouvrages sur le CANON X-07 , très demandés par les utilisateurs de cette superbe machine ...

Nous les avons composé avec grand plaisir et nous espèrons qu'ils auront apporté les réponses nécessaires aux nombreux problèmes auxquels se trouvent confrontés quotidiennement les canonistes .

Bien entendu , tout renseignement sur tel ou tel sujet traité dans "Les mystères du X-07" ou dans le présent ouvrage vous sera immédiatement délivré par le service technique du CLUB C7 : par courrier , par téléphone , de vive voix ou par MINITEL .

Les Editions NEPTUNE publieront régulièrement de nouveaux ouvrages informatiques en essayant toujours de garder une optique de bas prix , d'originalité et de haute qualité alliée à une simplicité de présentation .

N' hésitez surtout pas à nous contacter : le marché de l'informatique évolue rapidement et nos prochaines publications seront éditées avec l'intention certaine de rester à la pointe de l'actualité .

Bravo au X-07 qui nous a tellement étonné et qui nous surprendra sans doute encore longtemps ...

Le directeur de la Publication

André TONIC

ANNEXES

ANNEXE 1
ASS/DESASS

Maintenant que vous connaissez l'ASSEMBLEUR , il faudrait vous équiper d'un logiciel **"ASSEMBLEUR"** vous permettant d'échapper au studieux travail de rentrée des codes !!

Il existe pour le moment sur le marché deux ASSEMBLEURS très corrects : **celui de la société LOGI'STICK et celui du mensuel MICRO SYSTEMES** .

Le premier est relativement bien fait mais présente **quelques lacunes** : limitations au niveau des labels , pas de directives (ORG , DEFM ...) , un peu compliqué d'emploi .

Le deuxième , écrit entièrement en langage machine , est **pratiquement parfait** . En effet , il tient sur un carte mémoire de 4 Kilo-octets , possède beaucoup de pseudo-instructions ainsi que des fonctions de contrôle très puissantes . Nous le préférons nettement à celui de la société LOGI'STICK pour sa rapidité , sa compacité et , surtout , pour sa grande facilité d'emploi . Tous les listings source de cet ouvrage ont d'ailleurs été réalisés avec ce programme. Il est paru dans le numéro 49 de la revue MICRO SYSTEMES .

Vous devez sûrement vous douter qu'il existe un programme qui permet d'effectuer l'opération inverse d'un ASSEMBLEUR . Effectivement , quand vous avez assemblé un programme en mémoire , vous désirez probablement vérifier si cela a été bien opéré . Afin de calmer vos angoisses , il existe un logiciel appelé **"DESASSEMBLEUR"** permettant de lister la mémoire et de désassembler les codes entrés . En général , un DESASSEMBLEUR est accompagné d'un programme **"MONITEUR"** possèdant d'autres fonctions intéressantes (mode TRACE , sauvegarde de codes , listings de codes en ASCII , sortie sur imprimante ...) .

Trois logiciels se disputent la "suprématie du marché" : **celui de LOGI'STICK , de CANON et de MICRO SYSTEMES** . Nous allons systématiquement les passer en revue ...

La société LOGI'STICK édite , sur la même cassette que l'ASSEMBLEUR , un DESASSEMBLEUR-MONITEUR . A l'inverse du premier programme , le DESASSEMBLEUR est **très bien fait** et autorise de multiples opérations sur la mémoire . Il est rapide et très pratique d'utilisation . Actuellement , la cassette incluant l'ASSEMBLEUR et le DESASSEMBLEUR-MONITEUR est vendue au prix de **125 Francs par le CLUB C7** (Le plus bas prix constaté , réservé uniquement aux adhérents ...) .

La société CANON a commercialisé , sous forme de carte mémoire préprogrammée , un DESASSEMBLEUR-MONITEUR **très puissant** . Cette carte **XP-140F** , dotée de 8 kilo-octets de ROM et de 4 kilo-octets de RAM utilisateur, est vendue **entre 500 et 550 francs** . Elle permet quantité de manipulations grâce à dix fonctions BASIC supplémentaires (RENUM , MERGE , AUTO ...) et à un mode MONITEUR très performant (points d'arrêts , désassemblage multiples, entrée de codes ...) .

Enfin , le journal MICRO SYSTEMES a publié dans son numéro 42 , un DESASSEMBLEUR-MONITEUR **complet** . Il s'avère un peu moins fouillé que celui de la société LOGI'STICK mais possède tout de même une puissance de travail honnête .

Pour que vous n'ayez pas de problèmes à trouver ces divers logiciels , voici les adresses des différentes sociétés les commercialisant :

CANON FRANCE
7 av. Albert EINSTEIN
Centre d'affaire
93153 Le blanc Mesnil

MICRO SYSTEMES
2 à 12 rue de Bellevue
75940 Paris Cedex 19

LOGI'STICK
Centre d'affaire
"Le Bonaparte"
93153 Le Blanc Mesnil

ANNEXE 2
LE CLUB C7

Ces quelques lignes vont vous renseigner sur le CLUB C7 , association fondée en OCTOBRE 1984 . Elle a réussi à imposer **une gazette complète et très recherchée ainsi que des services de haute compétence** (permanence technique, stages , service courrier très développé ...) . Rejoindre le CLUB C7 , c'est entrer dans le monde fantastique des "Canonnistes" qui maîtrisent parfaitement leur machine préférée : **le CANON X-07** .

Le CLUB C7 : que du CANON X-07 ...

Le CLUB C7 est une association sous la loi 1901 . Son but est de faciliter la **compréhension** et le **développement** de l'ordinateur portable **CANON X-07** et d'agir comme un point de liaison entre les possesseurs du X-07 grâce à plusieurs atouts : une gazette , une programmathèque , une aide technique , une coopérative , des stages , etc ...

Le CLUB C7 a été fondé par des professionnels en informatique et des étudiants : toute personne désirant une aide , un journal , des contacts ... peut s'inscrire et bénéficier de tous les services du CLUB **gratuitement** , après avoir réglé sa cotisation **(380 francs pour l'année et 430 francs pour l'étranger).**

Le CLUB C7 : "le SON du CANON" ...

La gazette intitulée **"le SON du CANON"** a une vocation bimestrielle . Elle est constituée , en moyenne , de 40 pages format A4 traitant de tous les domaines concernant le X-07 : BASIC , LANGAGE MACHINE , PROCESSEURS, HARDWARE , PROFESSIONNEL , ROM , COURRIER , ADRESSES , ESSAIS de LOGICIELS ...

"Le SON du CANON" est envoyé aux adhérents avec **une K7 audio** récapitulant tous les programmes de ladite gazette : les efforts de frappe source d'erreurs , longs et pénibles sont ainsi évités .

Le CLUB C7 : L'AIDE TECHNIQUE ...

Cette AIDE TECHNIQUE se présente de 2 façons : les permanences et le courrier .

Les permanences téléphoniques se situent tous les mardi aprés-midi de **14h à 18h au 43.71.08.00** .

Néanmoins , vous pouvez passer nous voir le mardi après-midi uniquement à l'adresse du CLUB .

Nous fournissons aux adhérents , aussi bien par courrier que de vive voix, tous les renseignements dont nous disposons sur le CANON X-07 : **ROM désassemblée , cours d'apprentissage au BASIC et L.M. , informations complètes sur le HARDWARE , adresses , etc ...**

Cette aide est appuyée par les connaissances étendues de techniciens chevronnés travaillant sur .le X-07 depuis sa sortie en France : rien ne leur échappe !!

Le CLUB C7 : une PROGRAMMATHEQUE ...

Le développement de la programmathèque est lié à **la participation que lui apporte nos membres** : en effet , ce sont eux qui fournissent la majorité des programmes dont la liste est présente dans chaque gazette .

Pour encourager les programmeurs et garantir une bonne qualité des programmes , ceux-ci sont vendus à un prix modique (entre 5 et 20 Francs) et le programmeur recoit chaque année la moitié du montant total de la vente de ses programmes .

Les programmes commandés sont envoyés aux adhérents sur K7 avec une notice détaillée .

Le CLUB C7 : une COOPERATIVE ...

La coopérative a pour but principal de vendre des produits **moins chers que sur le marché** : nous disposons de beaucoup de produits et nous souhaitons un développement massif de cette section pour offrir aux adhérents le maximum d'avantages .

Pour tout renseignement complémentaire ou inscription , veuillez contacter le CLUB C7 au :
33 avenue Philippe AUGUSTE , 75011 PARIS
Tél : 43.71.08.00 (Métro NATION)

PUBLICATIONS C7

1

* L.M. & nombres aléatoires
* L'interface PERITEL X-720
* Kit appel, Maths,Astro de LSC
* Jeux : Magic Circus , Dactylo ...
* Trucs,astuces,informations ....
* BEEPS TRES SPECIAUX !!!



* Solitaire/Pentominos : L.M.
* Agenda (LSC) , Calc (P.S.)
* Camemberts Statistiques
* Tri de nombres en L.M.
* Anti-BREAK en L.M.
* Adresses : Mots-Clés ...

2

SPECIAL L.M.

* Auto-programmation,adresses
* PAINT : 3 fonctions graphiques
* FICHIER,ASS/DESASS de LSC
* Carte 40 Ko d'INFOSYSTEMES
* Jeu de la VIE en L.M.



* Traitement de textes (L.M.)
* adresses : fin des mots-clés
* Progs. en L.M. et BASIC
* Graphe (P.S.) , Forth (LSC)
* Références croisées (L.M.)
* Instruction BOX en L.M.



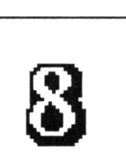
* Le sous-processeur T6834
* Les BOOLEENS du X-07
* Les sympathisants du X-07
* Nautilus,Calc,Graphe de LSC
* Les adresses : RST en folie
* COPYRIGHT en folie ...



* Essais : Jeux 2 et R.D.I.
* Microbox d'ADIRIS
* Les routines de conversion
* Labyrinthe 3D , alarme ...
* 40 programmes (ETSF)
* Inversion vidéo rapide .



* Essais : XFORTH , SAISIE 07
* La société E.R.I.E.
* Traceur multi-courbes .
* Notes de frais : pratique ...
* Le livre de l'année .
* Lecteur de disquettes .

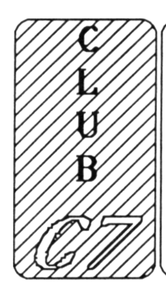


* Essais : X-740 , X-07 COM .
* Les sociétés de service .
* Polygones complexes .
* X-07 + LX-80 : génial !!
* Spécial X-720 : 2 logiciels .
* Télématique : feu vert !!



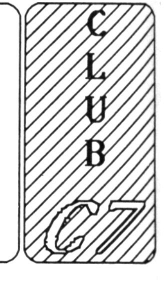
* Essai de la XP-150F .
* OLYMPIC GAMES 2 .
* Etiquettes : superbe !
* Vidéo : Rayon Z ...
* La télématique .
* Et des surprises !!





Si vous désirez acquérir l'un de ces numéros muni ou non de sa K7 (40 francs chaque n°) , veuillez contacter le service de la publication du CLUB C7 au : 33 avenue Philippe AUGUSTE 75011 PARIS (M° NATION) . Tél : 43.71.08.00

A BIENTOT !!



ANNEXE 3
"LES MYSTERES"

L' ouvrage tant attendu par les possesseurs du CANON X-07 est enfin arrivé !! En effet , depuis sa sortie en France , il y a de cela plus de deux ans (Début Novembre 1983) , aucun ouvrage n'avait traité de près ou de loin de **l' ASSEMBLEUR sur X-07** . Il est vrai que le sujet était relativement difficile d'accès ...

Devant la multitude de demandes émanant de toute la France , le CLUB C7 a décidé d'éditer un livre très complet divisé en trois parties :

_ La première partie permet de **se familiariser avec l'ASSEMBLEUR Z-80** utilisé par le CANON X-07 : idéal pour les néophytes !!

_ La seconde partie vous livre **tous les mystères du X-07** : architecture interne , ports de sortie , zone système , près de 150 adresses , la gestion des périphériques ...

_ La troisième partie est constituée par **huit applications pratiques** vous amenant progressivement à utiliser plusieurs routines-clés du X-07

Nous désirons souligner **l' aspect pédagogique** de ce livre qui n'a pas été négligé , loin de là : **plus de 40 schémas** , un style clair , des textes faciles à appréhender accompagnés de nombreux exemples ...

Nous souhaitons tous que cet ouvrage , destiné aussi bien aux débutants qu'aux programmeurs chevronnés , constitue l'aide définitive à leur essor vers **l'utilisation intensive de l'ASSEMBLEUR CANON** .

SPECIFICATIONS : 160 pages , format 21*14,5 , broché , couverture noire/jaune .

Si vous désirez vous procurer cet ouvrage , veuillez contacter Mr TONIC à l'un de ces numéros :
1/ CLUB C7 (Tél : 43.71.08.00) .
2/ Editions NEPTUNE (Tél : 64.56.81.56) .

ANNEXE 4
LA K7

QUOI ??? Mais c'est AHURISSANT !!!

Vous ne possédez pas encore la K7 des Editions NEPTUNE récapitulant tous les programmes importants de cet ouvrage ? ... Si vous avez un tant soit peu regardé les listings incorporés à ce livre , vous avez remarqué leur longueur assez imposante !

Alors n'hésitez plus !!!
Cette superbe K7 va vous éviter les pires tracas de l'informatique : l'entrée des codes !!

Elle est disponible pour la modique somme de 90 francs ... Où ça ???
Aux Editions NEPTUNE , bien sûr !!

Ecrivez-nous en joignant à votre courrier un chèque de 105 F
(90 F + 15 F de frais d'envoi)
à l'ordre des Editions NEPTUNE
et nous vous l'enverrons immédiatement !!

SOMMAIRE de la K7 :

1/ LMDATA (B)
2/ LOGO CANON (B)
3/ RESTORE (B)
4/ COPIE RAPIDE (B)
5/ LLIST (B+C)
6/ LOGOGENESE (B)
7/ LABYRINTHE (B)
8/ PENTOMINOS (B)
9/ SOLITAIRE (B)
10/ AUTONUM (B)
11/ REFBAS 16 Ko (B+C)
12/ EXABAS 16 Ko (B+C)
13/ LE PIEGE LCD (B+C)
14/ LE PIEGE VI. (B+C)
15/ OTHELLO (B+C)

Attention : stocks limités ...

ANNEXE 5
BIBLIOGRAPHIE

Cette bibliographie , non exhaustive , peut vous permettre de vous documenter sur tel ou tel aspect de l'ASSEMBLEUR Z-80 ou du CANON X-07 .

EDITIONS	TITRE
SYBEX	"La programmation du Z-80" de R. ZACKS
EYROLLES	"L'ASSEMBLEUR FACILE du Z-80"
P.S.I.	"Programmer en ASSEMBLEUR" de A. PINAUD
CANON	Manuel de la carte MONITEUR XP-140F
CLUB C7	Toutes les gazettes "Le SON du CANON"
CLUB C7	"LES MYSTERES DU X-07"
TESTS	L'Ordinateur Individuel
S.P.E.	Micro Systèmes

Cet ouvrage a été composé sur :





PROMOCOM
49 rue FONDARY
75015 PARIS
Tel : 45.79.80.12

Composé dans la continuité des « Mystères du X-07 », ce deuxième ouvrage complète les connaissances acquises lors de la lecture de ce dernier. Bâti autour de trois grandes parties (Soft, Technique, Applications), ce livre constitue la première référence en matière d'applications pures dans l'univers du X-07.

Tout en reprenant les recettes qui ont fait le succès des « Mystères » (taille, clarté, exemples, schémas...), « Applications en ASSEMBLEUR dans l'univers du CANON X-07 » apporte son lot de nouveautés avec une présentation originale, un style différent et surtout une pléiade d'applications qui satisferont le plus exigeant des canonistes.



Les Editions NEPTUNE
18 BIS, RUE VIOLET
75015 PARIS
TÉL. 64.56.81.56

APPLICATIONS EN ASSEMBLEUR DANS L'UNIVERS DU CANON X-07